Available online at www.sciencedirect.com
SCIENCE DIRECT®
Data & Knowledge Engineering xxx (2006) xxx–xxx
DATA & KNOWLEDGE ENGINEERING
www.elsevier.com/locate/datak

The phrase-based vector space model for automatic retrieval of free-text medical documents ☆

Wenlei Mao, Wesley W. Chu *

~~UCLA,~~ *Computer Science Department, University of California, 3731H Boelter Hall, Los Angeles, CA 90095-1596, United States*

Received 30 January 2006; accepted 2 February 2006

Abstract

Objective: To develop a document indexing scheme that improves the retrieval effectiveness ~~of~~ ^ free-text medical documents.

Design: The phrase-based vector space model (VSM) uses multi-word phrases as indexing terms. Each phrase consists of a concept in the unified medical language system (UMLS) and its corresponding component word stems. The similarity between concepts are defined by their relations in a hypernym hierarchy derived from UMLS. After defining the similarity between two phrases by their stem overlaps and the similarity between the concepts they represent, we define the similarity between two documents as the cosine of the angle between their corresponding phrase vectors. This paper reports the development and the validation of the phrase-based VSM.

Measurement: We compare the retrieval effectiveness of different vector space models using two standard test collections, OHSUMED and Medlars. OHSUMED contains 105 queries and 14,430 documents, and Medlars contains 30 queries and 1033 documents. Each document in the test collections is judged by human experts to be either relevant or non-relevant to each query. The retrieval effectiveness is measured by precision and recall.

Results: The phrase-based VSM is significantly more effective than the current gold standard—the stem-based VSM. Such significant retrieval effectiveness improvements are observed in both the exhaustive search and cluster-based document retrievals.

Conclusion: The phrase-based VSM is a better indexing scheme than the stem-based VSM. Medical document retrieval using the phrase-based VSM is significantly more effective than that using the stem-based VSM.

Keywords: Information storage and retrieval/methods; Computing methodologies; Vector space model; Concept-based vector space model; Phrase-based vector space model; Information systems; Unified medical language system

☆ Part of this research was presented in AMIA 2002 [1].

* Corresponding author. Tel.: +1 310 825 2047; fax: +1 310 825 2273.
E-mail address: ^wwc@cs.ucla.edu (W.W. Chu).

0169-023X/$ - see front matter © 2006 Elsevier B.V.. All rights reserved.
doi:10.1016/j.datak.2006.02.008

1. Introduction

Free-text documents are indispensable in medical practices. Medical literatures, patient records, and medical transcriptions according to doctors' dictations are some obvious examples. Computers are replacing pen and paper to become the major storage device and source of medical information. MEDLINE (medical literature, analysis, and retrieval system online), the US National Library of Medicine's (NLM) premier bibliographic database, contains over 12 million references to journal articles in biomedicine. It can now be easily accessed from the Web via either PubMed [2] or the NLM Gateway [3]. Electronic health record (EHR) has caught the attention of medical information technologists for over two decades now. Standards have been published by ASTM International [4], and are being addressed by the ISO Technical Committee 215. Despite many difficulties, medical institutes realize the importance of the EHR systems and are migrating from paper-based medical records to electronic records [5]. Furthermore, more and more patients, just like the physicians, have begun to use many of the medical resources on the Web such as MEDLINEplus [6].

The ever increasing amount of the medical text documents and the ever increasing dependence of people on such information require an effective document retrieval mechanism.

2. Background

2.1. The basics of current document retrieval systems

Document retrieval systems consist of two main processes, indexing and matching. Indexing is the process of selecting content identifiers to represent a text. Content identifiers are also called terms in this setting. Matching is the process of computing a measure of similarity between two text representations.

In some environments, human indexers assign terms selected from a controlled vocabulary. For example, JAMIA [7] suggests to use "three to ten key words or short phrases that will assist indexers. Terms from the medical subject headings list of *Index Medicus* are preferred." A more efficient alternative is to use automatic indexing where the system itself decides on the terms based on the text of the documents. A basic automatic indexing procedure for English might proceed as follows. First, divide the text into words; second, remove very frequent words such as prepositions and pronouns; and third, conflate related words to a common word stem by removing suffixes. The resulting word stems are used as the terms for the given text.

Since its inception, the vector space model (VSM) [8] is the most popular model in information retrieval. In this model, documents and queries are represented by vectors in a n-dimensional space, where n is the number of distinct terms. Each axis in this n-dimensional space corresponds to one term. Given a query, the system returns a ranked list of documents ordered by their similarities to the query. The problem of effective retrieval becomes the problem of returning documents relevant to the query first, so that the user spends less time sifting through non-relevant results. The similarity between a query and a document is often defined as the cosine of the angle between their respective vectors.

2.2. The problem

Although word stems have been shown to be quite effective indexing terms, a recurring question in document retrieval is: what should be used as the basic unit to identify the contents in the documents? Or, what is a term?

The problem of using word stems as terms is manifested in several ways:

1. The component words of a phrase sometimes have only remote, if any, relations with the phrase. For example, separating "photo synthesis" into "photo" and "synthesis" could be misleading.
2. Words could be too general. For example, the individual words "family" and "doctor" are not specific enough to distinguish between "family doctor" and "doctor family."
3. Different words could be used to represent the same thing. For example, both "hyperthermia" and "fever" indicate an abnormal body temperature elevation.

4. The same word could mean different things. For example, "hyperthermia" can indicate an abnormal body temperature elevation, as well as a treatment in which body tissue is exposed to high temperature to damage and kill cancer cells.

As a result, phrases and concepts were proposed to be used as content identifiers in place of words or word stems.

2.3. Phrases in document retrieval

In document retrieval, phrases are categorized into syntactic phrases and statistical phrases.

Syntactic phrases are those sets of words that satisfy certain syntactic relations. For example, if we specify that an adjective followed by a noun constitutes a phrase, then "high fever" is considered a phrase. Refs. [9–11] studied the use of syntactic phrases as content identifiers.

Statistical phrases are those word combinations that co-occur in a certain context in a text corpus more frequently than expected by chance. The following are some examples of statistical phrases: a pair of words that occur contiguously often enough [12]; a word pair that tends *not* to be separated by other words within the context of noun phrases [13,14]; and a set of n words that occurs in a sentence often enough [15], where n could take on several different values.

The effect of statistical phrases and syntactic phrases was compared in document retrieval [16,10,17]. Mitra et al. [17] observed that syntactic phrases performed better than statistical phrases when phrases were used alone as content identifiers, and the use of phrases did not significantly affect retrieval precision at the top ranks.

2.4. Concepts in document retrieval

Concepts are often encoded in controlled vocabularies such as dictionaries or thesauri, some of which are now conveniently available in electronic forms. The unified medical language system (UMLS) [18] is a popular controlled vocabulary for biomedical concepts.

Rada and Bicknell [19] used concepts in an older version of medical subject headings (MeSH) [20] as terms, defined the distance $\mathrm{dist}(t_i, t_j)$ between two terms t_i, t_j as the minimal number of broader-than edges between t_i and t_j, and defined the distance between a query q and a document d as

$$\mathrm{DISTANCE}(q,d) = \frac{1}{mn}\sum_{t_i \in q}\sum_{t_j \in d}\mathrm{dist}(t_i,t_j)$$

where m and n were the number of MeSH terms in the document and the query respectively. Six MeSH-encoded documents and ten encoded queries were ranked by the DISTANCE function and by two physicians. The agreement between DISTANCE and the human experts was found significant, while no significant correlation was observed if only exact matches between query terms and document terms were used in the document distance evaluation.

Hersh et al. [21] compared five different term selection schemes for document retrieval using three medical document test collections, each containing 200–2K abstracts, and 10–75 queries. The retrieval mechanism used corresponded to a weighted Boolean OR operation for all the query terms. The indexing terms used in the five methods were: (1) concepts in the Metathesaurus of UMLS, (2) words, (3) words that occurred in some UMLS concepts, (4) concepts and words that were not present in UMLS, and (5) concepts with their corresponding broader-than concepts in UMLS. The results showed that the word-based approaches (2–4) were much better than the concept-based approaches (1 and 5). There was no significant difference in the two word-based approaches (2 and 3) and the combination of words and concepts (case 4).

Yang and Chute [22] confirmed the results in [21] that when concepts in UMLS were used to represent documents, the retrieval performance was worse than when words were used. In two example-based approaches, human relevance judgments were used as training examples to derive word–concept and word–word correlations. Incoming word-based queries were then mapped to either concepts or words using the correlations

derived. Retrieval effectiveness improvement was observed for the example-based approaches over the no-learning word-based approach. They concluded that the empirical connections between different vocabularies used in the query and the documents learnt from the user judgments were more useful than those encoded in knowledge sources.

Many other attempts ~~of~~ using concepts in controlled vocabularies, such as WordNet [23], to replace word stems as terms in automatic document retrieval were also shown to be of little success [24–27].

Instead of using automatic indexing methods described above, Gonzalo et al. [28] showed that by manually tagging the queries and the documents with concepts from WordNet, they could improve the retrieval effectiveness significantly. Such a significant improvement indicated the potential of concept-based indexing. The poor performances of the other concept-based systems led us to the search of a better *automatic* retrieval system using concepts in a controlled vocabulary.

3. Vector space models

In the following sections, we shall use this example query from OHSUMED [29] to facilitate the discussion: "Hyperthermia, leukocytosis, increased intracranial pressure, and central herniation. Cerebral edema secondary to infection, diagnosis and treatment." The first part of the query is a brief description of the patient; the second part is the information ~~need.~~

Also, we shall discuss three types of schemes, the stem-based, concept-based, and phrase-based schemes, indicated by the superscripts, s, c and p respectively. We use s, r to denote stems, c, d to denote concepts, p, q to denote phrases, and x, y, z to denote documents.

3.1. Stem-based VSM

In the naivest approach, we could use words as the terms of the documents. Yet, morphological variants like "edema" and "edemas" are so closely related that they are usually conflated into a single word stem, e.g., "edem," by stemming. The two most popular stemmers are the Lovins stemmer [30] and the Porter stemmer [31]. The Lovins stemmer removes over 260 different suffixes using a longest-match algorithm. The Porter stemmer removes about 60 suffixes in a multiple-step approach; each step successively removes suffixes or transforms the stem. The Lovins stemmer produces word stems ("hypertherm"), ("leukocytos"), ("increas"), ("intracran"), ("pressur"), etc. for our example query.

Not all word stems in a document are equal. We use a stem weight to represent the relative importance of a word stem s in document x. The stem weights are generally computed following a *term frequency, inverse document frequency* (tf-idf) weighting scheme,

$$\tau_{s,x}\iota_s = \tau_{s,x}(\log_2 N/n_s + 1) \tag{1}$$

where $\tau_{s,x}$, the term frequency, is the number of times stem s appears in document x; and ι_s, the inverse document frequency of stem s, is determined by N (the number of documents in the collection) and n_s (the number of documents that contain stem s).

If we use S to represent the set of word stems in a document collection, then, we can model the documents as vectors in a $|S|$-dimensional space. Each base vector of the space corresponds to a word stem in S. We use a *stem vector*, x^s, to represent a document x, and define x^s as a set of ordered pairs $x^s = \{(s, \tau_{s,x})\}_{s\in S}$, where $\tau_{s,x}$ is the term frequency of stem s in document x. Furthermore, we define the *stem-based inner product*, $\langle x, y\rangle^s$, between documents x and y as

$$\langle x, y\rangle^s = \sum_{s\in S} \iota_s^2 \tau_{s,x}\tau_{s,y} \tag{2}$$

and define the *stem-based document similarity*, $\mathrm{sim}^s(x, y)$, between them as the cosine of the angle between the document vectors x^s and y^s,

$$\mathrm{sim}^s(x, y) = \frac{\langle x, y\rangle^s}{\sqrt{\langle x, x\rangle^s \langle y, y\rangle^s}}$$

In this stem-based document similarity definition, we assume that word stems are notational rather than conceptual entities; therefore, we treat different word stems as unrelated—there are no cross terms in the stem-based inner product (2).

3.2. Concept-based VSM

Using word stems to represent documents results in the inappropriate fragmentation of multi-word concepts such as "increased intracranial pressure" into their component stems like "increas," "intracran," and "pressur." Clearly, using concepts instead of single words or word stems as the terms should produce a vector space model that better mimics human thought processes, and therefore should result in more effective document retrieval.

However, the concept-based model is more complex than the stem-based model:

First, concepts are usually represented by multi-word phrases such as "increased intracranial pressure."

Second, there exist synonymous and polysemous phrases. A phrase is *polysemous* if it can be used to express different meanings. Two phrases are *synonymous* if they can be used to express the same meaning. For example, "fever" and "hyperthermia" are synonyms because both can be used to denote "an abnormal elevation of the body temperature." At the same time, "hyperthermia" is polysemous, because in addition to the above meaning, it can also be used to denote "a treatment in which body tissue is exposed to high temperature to damage and kill cancer cells." Synonyms can be identified with the help of a dictionary or a thesaurus. Determining which meaning a polysemous phrase represents is known as *word sense disambiguation* [32].

Third, some concepts are related to one another. Many semantic relations between concepts have been identified, the most well-known ones include hypernymy/hyponymy, and meronymy/holonymy relations [33,23]. A concept c is called a *hyponym* of another concept d if we say "A c is a (kind of) d." If c is a hyponym of d, then d is called a *hypernym* of c. Therefore, hypernymy/hoponymy are sometimes labelled as "is-a" relations. On the other hand, meronymy/holonymy are sometimes called "has-a" or "part-of" relations because we call c a *meronym* of d and d a *holonym* of c if we say "A c is a part of d," or "A d has a c (as a part)." For concrete examples, "fever" is a hyponym of "elevated body temperature," and "right upper lobe of lung" is a meronym of "lung."

Let us assume that we can partition documents into phrases for now. We shall ignore polysemy, and assume each phrase expresses just one concept. Concept identifiers are usually used to identify concepts. Using UMLS [18], our sample query becomes (C0015967), (C0023518), and (C0151740) etc., representing "hyperthermia," "leukocytosis," and "increased intracranial pressure," etc., respectively, where C0015967, C0023518, and C0151740 are *concept unique identifiers* (CUIs) in UMLS.

Just like in the stem-based VSM, we use a *concept vector* x^c to represent a document x, and define it as a set of ordered pairs $x^c = \{(c, \tau_{c,x})\}_{c \in C}$, where $\tau_{c,x}$ is the number of times concept c appears in document x, and C is the set of all concepts in the document collection. Furthermore, we define the *concept-based inner product*, $\langle x, y \rangle^c$, between documents x and y as

$$\langle x, y \rangle^c = \sum_{c \in C} \sum_{d \in C} \iota_c \tau_{c,x} \iota_d \tau_{d,y} s^c(c, d) \tag{3}$$

where $\iota_c, \iota_d > 0$ are the inverse document frequencies of concepts c and d respectively, and $s^c(c, d)$ quantifies the conceptual similarity between concepts c and d. The inverse document frequency of concept c is defined similar to the inverse document frequency of the stem s in Formula (1)

$$\iota_c = \log_2 N / n_c + 1$$

where n_c is the number of documents that contain concept c. We require the conceptual similarity $s^c(c, d)$ to be a symmetric function of concepts $c, d \in C$, $s^c(c, d) = s^c(d, c)$, that falls between 0 and 1 inclusively, $0 \leqslant s^c(c, d) \leqslant 1$, with a further constraint that $s^c(c, c) = 1$. Unlike in the stem-based inner product in Formula (2) where different stems are considered unrelated, we take the concept interrelations into consideration in the concept-based inner product (3). Using the concept-based document inner product, we again define the *concept-based document similarity* between documents x and y to be the cosine of the angle between their respective concept vectors,

$$\mathrm{sim}^c(x,y) = \frac{\langle x,y\rangle^c}{\sqrt{\langle x,x\rangle^c \langle y,y\rangle^c}}$$

3.3. Phrase-based VSM

Concepts in controlled vocabularies such as UMLS are used in the concept-based VSM. Conceptual similarities needed there are often derived from knowledge sources. The qualities of such vector space models therefore depend heavily on the qualities of the controlled vocabularies and the knowledge sources. Some concepts could be missing from the controlled vocabularies. For example, if we detect only concept C0021852 for "small bowel" in the phrase "infiltrative small bowel process" and find no concepts matching either the entire phrase, or the fragments "infiltrative" and "process," then we are losing important information when we represent documents using concepts only. Furthermore, missing certain conceptual relations in the knowledge sources potentially degrades retrieval effectiveness. For example, treating "cerebral edema" and "cerebral lesion" as unrelated is potentially harmful. Noticing the words "infiltrative" and "process" that match no concepts and the common component word "cerebral" in phrases "cerebral edema" and "cerebral lesion," we propose a phrase-based VSM to remedy the incompleteness of the controlled vocabularies and the knowledge sources.

In the phrase-based VSM, a document is represented as a set of phrases. Each phrase may correspond to multiple concepts (due to polysemy) and consist of several word stems. For example, "infiltrative small bowel process" is represented by phrases (; "infiltr"), (C0021852; "smal", "bowel"), (; "proces"). Our example query now becomes (C0015967, C0203597; "hypertherm"), (C0023518; "leukocytos"), and (C0151740; "increas", "intracran", "pressur") etc.

We use an ordered pair of two sets to represent a *phrase* $p = (\{(s,\pi_{s,p})\}_{s\in S}, \{(c,\pi_{c,p})\}_{c\in C})$. The first set, $\{(s,\pi_{s,p})\}_{s\in S}$, consists of ordered pairs that indicate the stems and their occurrence counts, $\pi_{s,p}$, in the phrase. The second set $\{(c,\pi_{c,p})\}_{c\in C}$ indicates the concepts and their occurrence counts, $\pi_{c,p}$, in the phrase. We denote the set of all phrases by P. Furthermore, we require that there is at least one stem in each phrase, i.e., for each phrase $p \in P$, there exists some stem s such that $\pi_{s,p} \geqslant 1$. We use a *phrase vector* x^p to represent a document x, $x^p = \{(p,\tau_{p,x})\}_{p\in P}$, where $\tau_{p,x}$ is the number of times phrase p occurs in document x. And we define the *phrase-based inner product* as

$$\langle x,y\rangle^p = \sum_{p\in P}\sum_{q\in P} \tau_{p,x}\tau_{q,y} s^p(p,q) \tag{4}$$

where we use $s^p(p,q)$ to measure the similarity between phrases p and q. We call $s^p(p,q)$ the *phrase similarity* between phrases p and q, and define it as

$$s^p(p,q) = \max\left(\left(f^s \sum_{s\in S} \imath_s^2 \pi_{s,p}\pi_{s,q}\right), \left(f^c \sum_{c\in C}\sum_{d\in C} \imath_c \pi_{c,p} \imath_d \pi_{d,q} s^c(c,d)\right)\right)$$

where $\imath_s, \imath_c, \imath_d > 0$ are the inverse document frequencies of stem s, concept c, and concept d respectively, and $s^c(c,d)$ is the conceptual similarity between concepts c and d. As in the concept-based VSM, we ignore polysemy and assume each phrase expresses only one concept,

$$\pi_{c,p} = \delta_{c,c_p} = \begin{cases} 1 & \text{if } c = c_p \\ 0 & \text{if } c \neq c_p \end{cases}$$

where c_p is the concept that phrase p expresses. Then the phrase similarity is reduced to

$$s^p(p,q) = \max\left(\left(f^s \sum_{s\in S} \imath_s^2 \pi_{s,p}\pi_{s,q}\right), \left(f^c \imath_{c_p} \imath_{d_q} s^c(c_p,d_q)\right)\right) \tag{5}$$

where c_p is the concept phrase p expresses, and d_q is the concept q expresses. Here we use two contribution factors, f^s and f^c, to specify the relative importance of the stem contribution and the concept contribution in the overall phrase similarity. The stem contribution

$$f^s \sum_{s \in S} \iota_s^2 \pi_{s,p} \pi_{s,q}$$

measures the stem overlaps between phrases p and q, and the concept contribution

$$f^c \iota_{c_p} \iota_{d_q} s^c(c_p, d_q)$$

takes the concept interrelation into consideration. ~~Conceptually, w~~hen combining the stem contribution and the concept contribution this way, we use stem overlaps to compensate for the incompleteness of the controlled vocabularies in encoding all necessary concepts, and the incompleteness of the knowledge sources in describing all necessary concept interrelations. Once again, we define the *phrase-based document similarity* between documents x and y to be the cosine of the angle between their respective phrase vectors,

$$\mathrm{sim}^p(x,y) = \frac{\langle x, y \rangle^p}{\sqrt{\langle x, x \rangle^p \langle y, y \rangle^p}} \tag{6}$$

4. Methods

4.1. Conceptual similarity evaluation

In this paper, we concentrate on the hypernym relations and derive the conceptual similarity between a pair of ancestor-descendant concepts in a hypernym hierarchy based on the following observations:

1. Two concepts closer together in a hypernym hierarchy are more closely related to one another than those farther apart.
2. Specific concepts are conceptually more strongly related to one another than general ones. We could use the number of descendants of a concept to measure its generality.
3. Consider two concepts c_0 and d_0, where c_0 is the only direct hypernym of d_0, d_0 the only hyponym of c_0, and d_0 has no hyponym of its own. Concepts c_0 and d_0 are so much alike that we define the conceptual similarity between them to be 1.

As a result, we define the conceptual similarity between a pair of ancestor-descendant concepts c and d in a hypernym hierarchy as

$$s^c(c,d) = \frac{1}{l(c,d)\log_2(D(c) + D(d) + 1)} \tag{7}$$

where $l(c,d)$ is the hierarchy distance between c and d, and $D(c), D(d)$ are the descendant counts for c, d respectively. We further define $s^c(c,c) = 1$ for all concepts. Based on the observations that $l(c,d) = l(d,c) \geqslant 1$, $D(c) \geqslant 0$, $D(d) \geqslant 0$, and at least one of $D(c)$ and $D(d)$ is no less than 1, it is not difficult to see the conceptual similarity thus defined satisfies the requirements in the concept-based VSM and the phrase-based VSM: $0 \leqslant s^c(c,d) \leqslant 1$, $s^c(c,d) = s^c(d,c)$, and $s^c(c,c) = 1$.

4.2. The knowledge source, UMLS

UMLS [18] is a medical lexical knowledge source and a set of associated lexical programs. The knowledge source consists of the UMLS Metathesaurus, the SPECIALIST lexicon, and the UMLS semantic network. Particularly of interest to us is its central vocabulary component—the Metathesaurus. It contains 1.6M biomedical phrases representing over 800K concepts from more than 60 vocabularies and classifications.

A concept unique identifier (CUI) identifies each concept. Because of synonymy, multiple phrases can be associated with one CUI. For example, 71 phrases in 15 languages are associated with CUI C0015967. Some example English phrases for that CUI are "fever," "high body temperature," "temperature, high," and "hyperthermia." On the other hand, a phrase can express multiple meanings. For example, "hyperthermia" can be associated with both C0015967 (the "fever" sense) and C0203597 (the "treatment" sense).

Table 1
Comparison of OHSUMED and Medlars statistics

	OHSUMED		Medlars	
	Query	Document	Query	Document
Number of documents	*105*	*14,430*	*30*	*1033*
Phrases per document	*7.5*	112	*11*	*90*
Stems per phrase	1.34	1.25	1.25	1.14
Concepts per phrase	1.21	1.18	1.27	1.21
Multi-stem phrases per document	1.96	*21.3*	2.6	*10.8*
Multi-sense phrases per document	*1.2*	11.3	*2*	9.8

Noticeable differences are shown in italic fonts.

The Metathesaurus encodes many conceptual relations. We are particularly interested in the hypernym/hyponym relations. Two pairs of relations in UMLS roughly correspond to the hypernym/hyponym relations: the RB/RN (border than/narrower than) and the PAR/CHD (parent/child) relations. For example, C0015967 (fever) has a parent concept C0005904 (body temperature change). RB and RN are redundant—for two concepts c and d, if (c, d) is in the RB relations, then (d, c) is in the RN relations, and vice versa. Similarly, PAR and CHD are redundant. As a result, we combine RB and PAR into a single hypernym hierarchy. Hypernymy is transitive [34]. For example, ''sign and symptom'' is a hypernym of ''body temperature change,'' and ''body temperature change'' a hypernym of ''hyperthermia,'' so ''sign and symptom'' is also a hypernym of ''hyperthermia.'' However, the UMLS Metathesaurus encodes only the direct hypernym relations but not the transitive closure. We derive the transitive closure of the hypernym relation and use Formula (7) to compute the conceptual similarities.

UMLS plays two important roles in the concept-based VSM and the phrase-based VSM. First, we use its Metathesaurus as a controlled vocabulary in phrase detection. Second, we use the hypernym relations encoded in RB and PAR in conceptual similarity derivation.

4.3. The test collections

To compare the effectiveness of different vector space models in document retrieval, we need a test collection that provides (1) a set of queries, (2) a set of documents, and (3) the judgments indicating if a document is relevant to a query.

OHSUMED [29] is a test collection widely used in recent information retrieval tests. OHSUMED contains 106 queries. Each query contains a patient description and an information need. Our example ~~query~~ is query 57 in the collection. The document collection is a subset of 348K MEDLINE references from 1987 to 1991. Seventy-five percent of the references contain titles and abstracts, while the remainder have only titles. Each reference also contains human-assigned subject headings from the medical subject headings. References (14,430) in the document collection are judged by ''physicians who were clinically active and were current fellows in general medicine or medical informatics or senior medical residents'' to be definitely relevant, possibly relevant, or non-relevant to each of the 105[1] queries. The standard recall and precision evaluation that we shall discuss later requires a binary relevance judgment—relevant or non-relevant. This can be easily achieved by merging the definitely relevant and the possibly relevant documents into a single relevant category.

Another test collection Medlars [35] is based on MEDLINE references collections from 1964 to 1966. It has been used extensively in document retrieval system comparisons. There are 30 queries and 1033 references in the collection. The judgments are provided by ''a medical school student.''

We use both test collections to compare the retrieval effectiveness of different methods. However, based on the qualification of the human experts, the extent, and the up-to-dateness of these collections, we believe that OHSUMED reflects expert judgment better; therefore we direct the attention of the reader to the results obtained from OHSUMED collection in later sections. Table 1 compares some statistics of the two collections.

[1] One query has no relevance judgments.

Besides the collection size difference discussed above, other noticeable differences include: OHSUMED queries are slightly shorter than those in Medlars; OHSUMED documents on average contain more long phrases (those with more than one stems); and Medlars contains slightly more polysemous phrases (those with multiple senses).

4.4. Phrase detection

The building blocks of the concept-based VSM and the phrase-based VSM are phrases. A phrase usually consists of multiple words. Given a controlled vocabulary containing a set of phrases, P, and a set of documents, X, we need to efficiently detect the occurrences of the phrases in P in each of the documents in X. A naive algorithm (see [36]) requires $O(N_x N_p)$ word comparisons in the worst case, where N_x is the total number of words in the document set X and N_p is the total number of words in all the phrases in P. There are $N_p = 6.7$M words in the 1.3M English phrases in UMLS. Using the statistics of the larger OHSUMED collection shown in Table 1, we see that on average there are $112 \times 1.25 \times 14\text{K} = 2.0$M words in the test documents. The naive algorithm described above is too time consuming and thus unacceptable for phrase detection. On the other hand, the Aho–Corasick algorithm [37] detects all the occurrences of the phrases in P from the documents in X using $O(N_x + N_p)$ word comparisons. Therefore, we adapt the Aho–Corasick algorithm for phrase detection:

1. The Aho–Corasick algorithm detects all occurrences of any phrase in a document. However, we only keep the longest, most specific phrase. For example, although both "edema" and "cerebral edema" are detected in the sample query, we keep only the latter, more specific concept, and ignore the former, more general concept.
2. To detect multi-word phrases, we match stems instead of words in a document with the UMLS phrases. To avoid conflating different abbreviations into a single stem, we define the stem for a word shorter than four characters to be the original word.
3. In English, about 250 common words such as "a" and "the" appear very frequently. It is a standard practice to include them in a stop list and remove them from document representations [38]. In our phrase detection, we remove the stop words in the stop list *after* the multi-word phrase detection. In this way, we correctly detect "secondary to" and "infection" from "cerebral edema secondary to infection." We would incorrectly detect "secondary infection" if the stop words ("to" in this case) were removed before the phrase detection.

Polysemy is one of the difficulties people encounter when using concepts. A polysemous phrase can express multiple meanings. As a result, it is necessary to disambiguate polysemous phrases in document retrieval. For example, seeing "hyperthermia," it is necessary to figure out whether it means "fever" or a type of "treatment" by word sense disambiguation [32]. The current accuracy and efficiency of word sense disambiguation algorithms are low. We perform a very primitive word sense disambiguation based on the following observation. UMLS tends to assign a smaller CUI to the more popular sense of a phrase. For example, the CUI for the "fever" sense of "hyperthermia" is C0015967, while the CUI for its "treatment" sense is C0203597. Therefore, we use the concept corresponding to the smallest CUI in the concept-based VSM and the phrase-based VSM.

4.5. Retrieval effectiveness measures

The goal of document retrieval is to return documents relevant to a user query before non-relevant ones. The effectiveness of a document retrieval system is measured by the recall and precision [39,40] based on the user's judgment of whether each document is relevant to a query q. When a certain number of documents are returned, we define *precision* to be the proportion of the retrieved documents that are relevant; and define *recall* to be the proportion of the relevant documents retrieved so far. More specifically, if we use R_q to represent the set of documents relevant to q, and A to represent the set of retrieved documents, then we define

$$\text{precision} = \frac{|R_q \cap A|}{|A|} \quad \text{and} \quad \text{recall} = \frac{|R_q \cap A|}{|R_q|} \tag{8}$$

There are several ways to evaluate the retrieval effectiveness using recall and precision.

To visually display the change in the precision values as documents are retrieved, we interpolate the precision values to a set of 11 recall points $0, 0.1, 0.2, \ldots, 1$. Averaging the precision values over a set of queries at these recall points illustrates the behavior of a system. Further averaging the 11 average precision values, we arrive at the *average 11-point average precision*, denoted by $\mathscr{G}_{P_{11}}$. Instead of interpolating the precision values to a set of standard recall points, we could also compute the average precision values after each relevant document is retrieved. The average of such a value over a set of queries is called the *average precision*, denoted by $\mathscr{G}_P$.

The two retrieval effective measures, $\mathscr{G}_{P_{11}}$ and $\mathscr{G}_P$, described above measure the average retrieval effectiveness of a system when different amount of documents are retrieved. Sometimes, it is important to know the performance of a system after a certain number of documents are retrieved. We use the *average precision at cutoff level*, $\mathscr{G}_{P_{\gamma=n}}$, to measure the average of the precision values over a set of queries when n documents are retrieved. Similarly, we use the *average recall at cutoff level*, $\mathscr{G}_{R_{\gamma=n}}$, to measure the average of the recall values when n documents are retrieved. By varying the cutoff level n, we can study the effectiveness of a system using two families of such measures.

$\mathscr{G}_{P_{\gamma=n}}$ and $\mathscr{G}_{R_{\gamma=n}}$ describe the performance of a system when a fixed number of documents are retrieved. We could also study the performance of a system when some query-specific condition is satisfied. Let us use R_q to denote the set of documents relevant to query q, and $|R_q|$ the number of documents relevant to query q. The *average precision at* $|R_q|$, $\mathscr{G}_{P_{|R_q|}}$, measures the average of the precision values when $|R_q|$ documents are retrieved over a set of queries. The *average precision at half recall*, $\mathscr{G}_{P_{.5}}$, on the other hand, measures the average precision values when half of the relevant documents have been retrieved.

5. Results

5.1. Comparison of the recall–precision curves

Figs. 1 and 2 depict the average precision values of 105 OHSUMED queries and 30 Medlars queries, respectively, at the 11 standard recall points $0, 0.1, 0.2, \ldots, 1$ for five different vector space models. For the OHSUMED results,

1. "Stems" is the baseline generated by the stem-based VSM. Its average 11-point average precision is $\mathscr{G}^s_{P_{11}} = 0.376$.
2. "Concepts Unrelated" is generated by using the concepts as the terms, and treating different concepts as unrelated. More specifically, we use $s^c(c,d) = \delta_{c,d}$ in the inner product calculation (Formula (3)). The average 11-point average precision is $\mathscr{G}^{cu}_{P_{11}} = 0.336$, an 11% decrease from the baseline.
3. "Concepts:" Similar to case 2, but taking the concept interrelations into consideration, we achieve a significant improvement over case 2. The average effectiveness is approximately equal to that of the baseline.
4. "Phrases, Concepts Unrelated:" Considering contributions from both the concepts and the word stems in a phrase, but once again, treating different concepts as unrelated by setting $s^c(c_p, d_q)$ in Formula (5) to δ_{c_p,d_q}, we achieve significant improvement over the "Concept Unrelated" case. In fact, its average 11-point average precision is $\mathscr{G}^{pcu}_{P_{11}} = 0.403$, 7.1% better than the baseline.
5. "Phrases:" Similar to case 4, but considering the concept interrelations, we achieve an average 11-point average precision of $\mathscr{G}^{p}_{P_{11}} = 0.433$, which is a significant 15% improvement over the baseline. In both cases 4 and 5, we used equal weight for the stem and the concept contributions, $f^s = f^c = 1$.

Our experimental results reveal that using only concepts to represent documents and treating different concepts as unrelated can cause the retrieval effectiveness to deteriorate (case 2). Considering the concept interrelations (case 3) or relating different phrases by their shared word stems (case 4) can both improve retrieval effectiveness. Measuring the similarity between two phrases using their stem overlaps and the relation between the concepts they represent, the phrase-based VSM (case 5) is significantly more effective than the stem-based VSM.

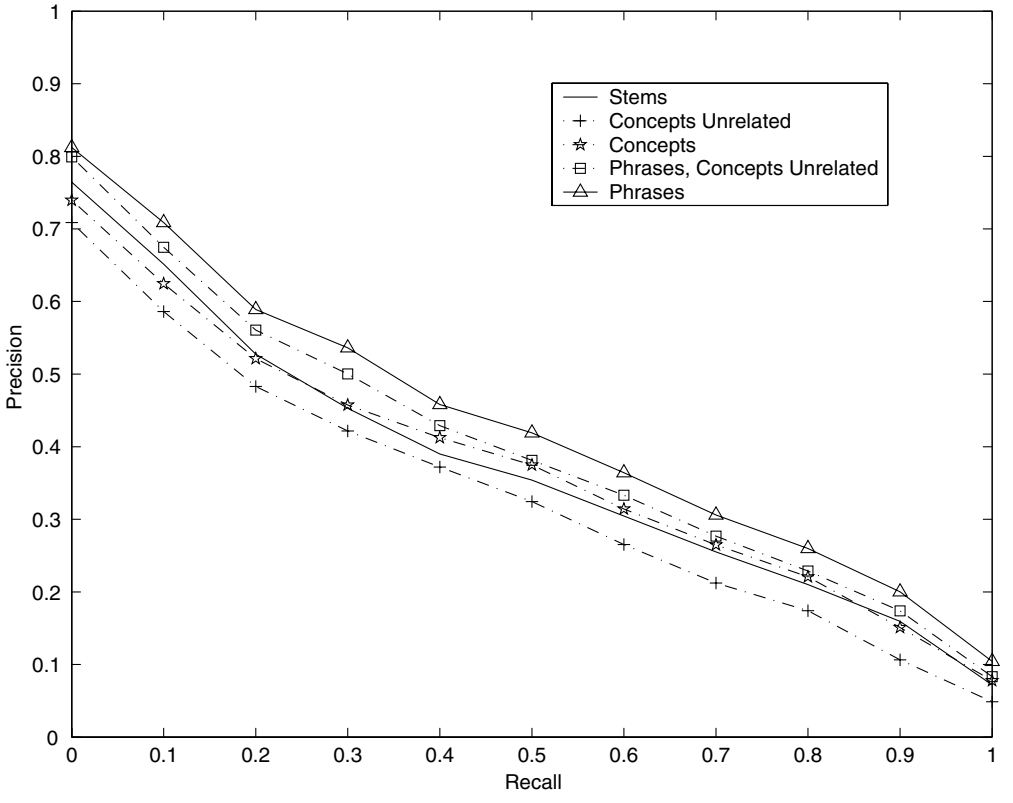

Fig. 1. Comparison of the average recall–precision curves over 105 OHSUMED queries.

5.2. Sensitivity of retrieval effectiveness to f^s and f^c

To generate the two sets of recall–precision curves "Phrase, Concept Unrelated" and "Phrase" in Figs. 1 and 2, we used equal weight, $f^s = f^c = 1$. To study the relative importance of the stem contribution and the concept contribution in the inner product calculation, we vary the weights f^s and f^c and study the change of the average 11-point average precision value $\mathscr{G}_{P_{11}}$. From Formulae (4)–(6), it is easy to see that the document similarity value depends on the ratio between f^s and f^c, not their absolute values; therefore, we vary the (f^s, f^c) from the stem-only case $(1, 0)$, to the equal-weight phrase case $(1, 1)$, to the concept-only case $(0, 1)$, and study the change of the average 11-point average precision values.

Fig. 3 depicts the changes of the average 11-point average precision values as the result of the change of f^s and f^c. We observe that the retrieval effectiveness measured by $\mathscr{G}_{P_{11}}$ is maximized when f^c is about the same as f^s, and, in this region, the retrieval effectiveness is not sensitive to the change of the relative importance of the stem contribution and the concept contribution.

5.3. Summary of retrieval effectiveness values

Tables 2 and 3 contain the retrieval effectiveness values for OHSUMED and Medlars respectively. To save space, we abbreviate the names of the methods using S for "Stems," CU for "Concepts Unrelated," C for "Concepts," PCU for "Phrases, Concept Unrelated," and P for "Phrases." For each effectiveness value in the CU, C, PCU, or P cases, we list its percentage difference from its corresponding baseline S value under the symbol (±%). Buckley and Voorhees [41] pointed out that a 5% difference in average precision over 50 queries usually indicates the difference between two systems. Therefore, we see from the results that only considering concepts in the queries and the documents is not enough, even if concept interrelations are taken into

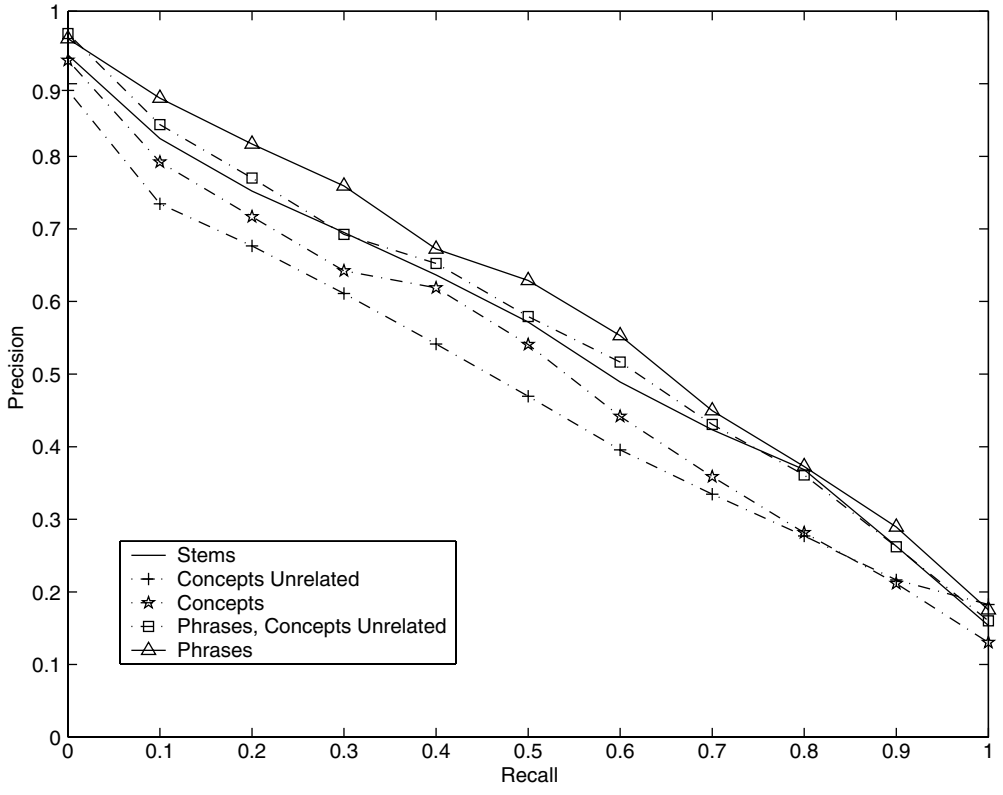

Fig. 2. Comparison of the average precision-recall over 30 Medlars queries.

account. Considering the stem overlaps among the phrases of the two documents improves the retrieval effectiveness. Significant retrieval effectiveness improvements from the stem-based VSM are achieved when both the stem overlaps and the conceptual similarities between related concepts are considered.

In addition to the average percentage difference, we test the significance of the results to see if the difference observed between each of the CU, C, PCU, and P values and the baseline S value could come from sampling errors. To compare two document similarity measures, say P versus S, we first select an effectiveness measure, say, the precision at cutoff $\chi = 10$. Then we compute the precision value for each query q using the P and the S method, and denote the results as $\pi^{P}_{q,\chi=10}$ and $\pi^{S}_{q,\chi=10}$ respectively. Usually, we observe the difference $\pi^{P}_{q,\chi=10} - \pi^{S}_{q,\chi=10}$ to be positive for some queries and negative for others. The +7.9% difference registered in row $\mathscr{G}_{P_{\chi=10}}$ under column P of Table 2 is an aggregate of such differences. To claim that such an improvement occurs not by chance, we perform t-test on the differences. First, we set up a null hypothesis stating that the difference between methods P and S, $\pi^{P}_{q,\chi=10} - \pi^{S}_{q,\chi=10}$, has a zero mean. Then, we set up two alternative hypotheses: (1) method P is better in the sense that the difference is positive; and (2) method S is better. To reject the null hypothesis in favor of either of these two alternatives, we perform t-test over a set of queries using MATLAB. A greater-than (>) or a less-than symbol (<) in Tables 2 and 3 indicate that there are significant evidences (with a confidence level of at least 95%) that the method under consideration is either better than or worse than the baseline "Stems" method, respectively. A question mark (?), on the other hand, indicates the lack of significant evidences. If there is enough evidence that one method is better or worse than the baseline, we also list the significance value "sig" to indicate the probability that the conclusion is arrived at *by chance* under the null hypothesis. A lower "sig" value indicates a higher confidence.

5.4. Retrieval effectiveness comparison in cluster-based document retrieval

In the previous sections, we showed that the phrase-based VSM is more effective than the stem-based VSM in document retrieval using exhaustive search. Let us consider a set of N documents. In an exhaustive search

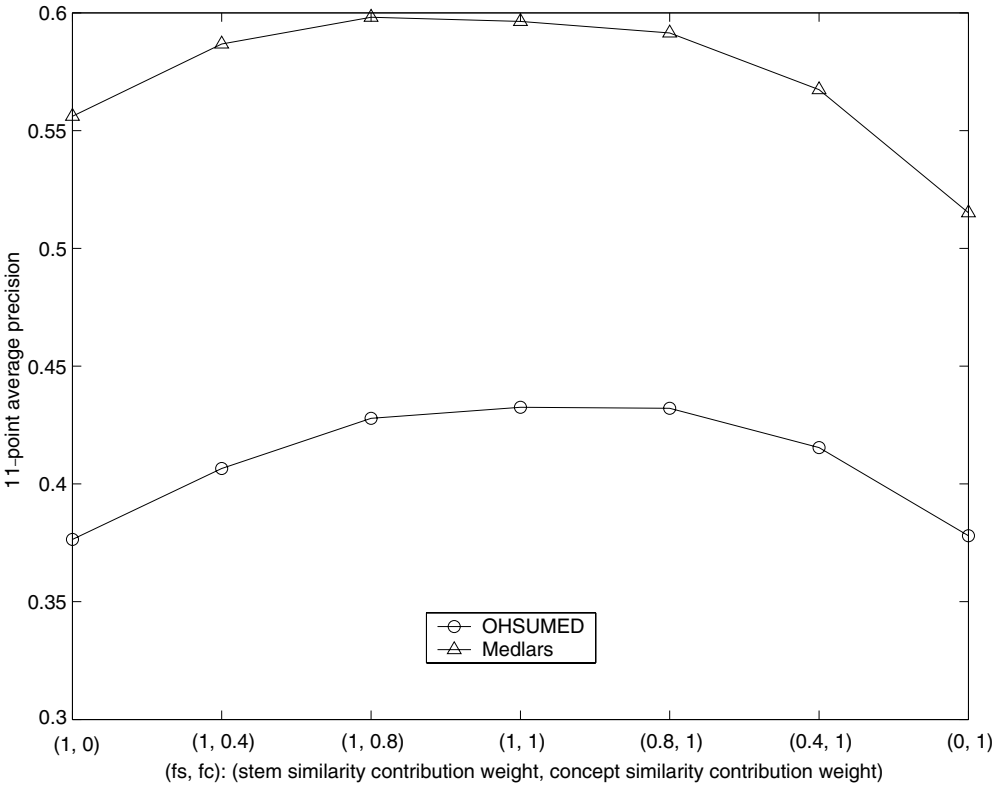

Fig. 3. Sensitivity of $\mathcal{G}_{P_{11}}$ to f^s, f^c changes in OHSUMED and Medlars.

Table 2
Comparison of the retrieval effectiveness values in OHSUMED

	S	CU ±%	CU?S, sig	C ±%	C?S, sig	PCU ±%	PCU?S, sig	P ±%	P?S, sig
$\mathcal{G}_{P_{11}}$	0.376	0.336 −11	<, 0.01	0.378 +0.5	?	0.403 +7.1	>, 3×10^{-4}	0.433 +15	>, 5×10^{-8}
$\mathcal{G}_{P}$	0.359	0.318 −11	<, 8×10^{-3}	0.363 +1.1	?	0.386 +7.5	>, 6×10^{-4}	0.416 +16	>, 7×10^{-8}
$\mathcal{G}_{P_{\chi=2}}$	0.595	0.567 −4.7	?	0.590 −0.8	?	0.657 +10	>, 0.02	0.662 +11	>, 0.02
$\mathcal{G}_{P_{\chi=10}}$	0.483	0.456 −5.6	?	0.480 −0.6	?	0.510 +5.6	>, 9×10^{-3}	0.521 +7.9	>, 6×10^{-3}
$\mathcal{G}_{P_{\chi=20}}$	0.410	0.409 −0.2	?	0.412 +0.5	?	0.435 +6.1	>, 3×10^{-4}	0.447 +9.0	>, 4×10^{-5}
$\mathcal{G}_{P_{\chi=100}}$	0.252	0.231 −8.3	<, 5×10^{-3}	0.250 −0.8	?	0.263 +4.4	>, 2×10^{-5}	0.274 +8.7	>, 2×10^{-6}
$\mathcal{G}_{R_{\chi=10}}$	0.153	0.133 −13	?	0.148 −3.2	?	0.167 +9.2	>, 0.03	0.172 +12	>, 9×10^{-3}
$\mathcal{G}_{R_{\chi=20}}$	0.236	0.231 −2.1	?	0.231 −2.1	?	0.255 +8.1	>, 4×10^{-3}	0.262 +11	>, 2×10^{-3}
$\mathcal{G}_{R_{\chi=100}}$	0.586	0.530 −10	<, 2×10^{-3}	0.573 −8.4	?	0.609 +3.9	>, 2×10^{-3}	0.647 +10	>, 9×10^{-7}
$\mathcal{G}_{R_{\chi=200}}$	0.745	0.659 −12	<, 3×10^{-5}	0.738 −0.9	?	0.767 +3.0	>, 5×10^{-3}	0.812 +9.0	>, 7×10^{-7}
$\mathcal{G}_{P_{\lvert R_q\rvert}}$	0.365	0.333 −8.8	<, 0.03	0.367 +0.5	?	0.388 +6.3	>, 4×10^{-3}	0.410 +12	>, 1×10^{-5}
$\mathcal{G}_{P.5}$	0.347	0.318 −8.4	?	0.369 +6.3	?	0.375 +8.1	>, 7×10^{-3}	0.412 +19	>, 5×10^{-5}

system, the similarity values between an incoming query and all the N documents need to be computed *online* before the documents can be returned to the user. Because of the relatively large computation complexity of the vector space models, such an exhaustive search scheme is not feasible for large document collections. Using hierarchical clustering algorithms, we can first construct a document hierarchy using $O(N\log N)$ *offline* document similarity computations, and return a ranked list of documents using only $O(\log N)$ online comparisons.

We compare the stem-based VSM and the phrase-based VSM using an $O(N\log N)$ spherical k-means algorithm that has been shown to produce good clusters in document clustering [42,43]. The resulting document clusters are searched using top-down and bottom-up searching strategies. Fig. 4 contains the recall–precision curves of six different searching strategies on the OHSUMED data.

Table 3
Comparison of the retrieval effectiveness values in Medlars

	S	CU ±%	CU?S, sig	C ±%	C?S, sig	PCU ±%	PCU?S, sig	P ±%	P?S, sig
$\mathcal{G}_{P_{11}}$	0.556	0.484 −13	<, 0.05	0.515 −7.3	?	0.567 +2.0	?	0.596 +7.2	>, 0.02
$\mathcal{G}_{P}$	0.533	0.447 −16	<, 3×10^{-3}	0.498 −6.6	?	0.550 +3.2	>, 0.05	0.581 +9.0	>, 7×10^{-3}
$\mathcal{G}_{P_{\chi=2}}$	0.783	0.667 −15	?	0.733 −6.4	?	0.817 +4.3	?	0.783 +0	?
$\mathcal{G}_{P_{\chi=10}}$	0.609	0.543 −11	<, 0.04	0.613 +0.7	?	0.647 +6.2	>, 0.02	0.673 +11	>, 8×10^{-3}
$\mathcal{G}_{P_{\chi=20}}$	0.535	0.443 −17	<, 0.02	0.497 −7.1	?	0.552 +3.2	?	0.578 +8.0	>, 0.03
$\mathcal{G}_{P_{\chi=100}}$	0.196	0.186 −5.1	?	0.181 −7.7	?	0.198 +1.0	?	0.203 +3.6	?
$\mathcal{G}_{R_{\chi=10}}$	0.295	0.257 −13	<, 9×10^{-3}	0.293 −0.7	?	0.312 +5.8	>, 0.02	0.323 +9.5	>, 2×10^{-3}
$\mathcal{G}_{R_{\chi=20}}$	0.497	0.414 −17	<, 5×10^{-3}	0.456 −8.2	?	0.512 +3.0	?	0.537 +8.0	>, 0.02
$\mathcal{G}_{R_{\chi=100}}$	0.854	0.800 −6.3	?	0.799 −6.4	?	0.863 +1.1	?	0.883 +3.4	?
$\mathcal{G}_{R_{\chi=200}}$	0.915	0.886 −3.2	?	0.886 −3.2	?	0.930 +1.6	>, 0.04	0.944 +3.2	>, 0.03
$\mathcal{G}_{P_{\lvert R_q\rvert}}$	0.523	0.418 −20	<, 2×10^{-3}	0.496 −5.2	?	0.538 +2.9	>, 0.04	0.556 +6.3	>, 0.03
$\mathcal{G}_{P_{.5}}$	0.552	0.441 −20	<, 7×10^{-3}	0.528 −4.3	?	0.569 +3.1	?	0.614 +11	>, 9×10^{-3}

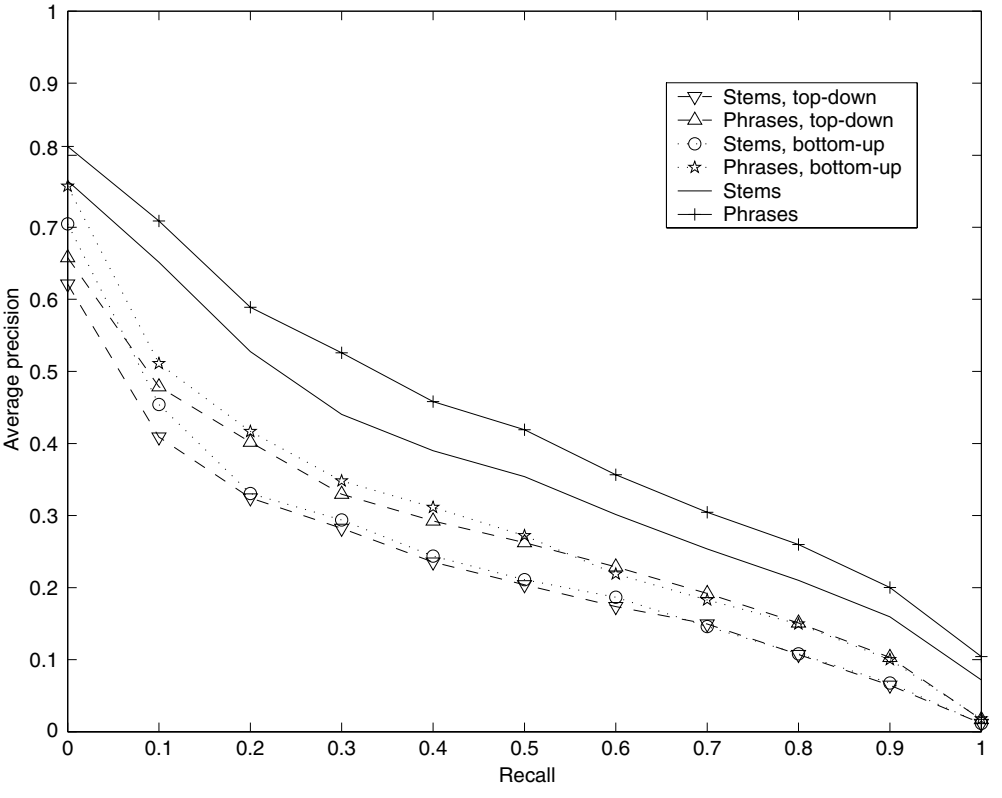

Fig. 4. Retrieval effectiveness comparison in OHSUMED.

The two curves "Stems" and "Phrases" are extracted from Fig. 1. They are the result of an exhaustive search on the 14K documents in OHSUMED. Their average 11-point average precision values are

$$\mathcal{G}^{s}_{P_{11}} = 0.376 \quad \text{and} \quad \mathcal{G}^{p}_{P_{11}} = 0.433$$

The other four curves depict the retrieval effectiveness of systems when the document hierarchies are searched. Clearly, the retrieval effectiveness of the cluster-based approaches is lower than that of the exhaustive-search-based approaches. That is, by using a cluster-based document retrieval, we sacrifice the retrieval effectiveness for more efficient retrieval. More importantly, using the same searching strategy, we see that the retrieval effec-

tiveness of the phrase-based VSM is always much better than that of the stem-based VSM. For the top-down search,

$$\mathscr{G}^{s,td}_{P_{11}} = 0.235 \quad \text{and} \quad \mathscr{G}^{p,td}_{P_{11}} = 0.283$$

and for the bottom-up search,

$$\mathscr{G}^{s,bu}_{P_{11}} = 0.251 \quad \text{and} \quad \mathscr{G}^{p,bu}_{P_{11}} = 0.299$$

In each case, the phrase-based VSM is about 20% more effective than the stem-based VSM.

5.5. Computation complexity

The document similarity calculation in the phrase-based VSM is more complex than that in the stem-based VSM. Let us use L to represent the average length of a document. In the stem-based VSM, different word stems are considered unrelated. As a result, by building indexes on the word stems in the documents, an efficient algorithm computes the stem-based similarity between two documents using $O(L \log L)$ time. The time complexity of a straightforward implementation of the phrase-based document similarity calculation is $O(L^2)$. Different phrases in the phrase-based VSM can be related to one another not only because they may share common word stems, but also because the concepts they represent can be related. Therefore, indexing on the phrases in the documents does not reduce the time complexity of the phrase-based document similarity calculation to $O(L \log L)$. To reduce the computation complexity, we need to build separate indexes on the concepts and the stems in the documents, keep track of where each stem or concept occurs, and modify the conceptual similarity storage structure. The phrase-based document similarity calculation utilizing such data structure modifications has an $O(L \log L)$ time complexity. For the OHSUMED documents, the improved phrase-based document similarity calculation is about 10 times slower than the stem-based calculation, while the straightforward implementation is over 250 times slower than the stem-based calculation.

Preliminary experimental results show that the number of related concept pairs decreases drastically as the pairwise conceptual similarity value increases. Therefore, we can further reduce the phrase-based computation complexity by treating related concepts with low conceptual similarity values as unrelated. We are currently investigating the tradeoff between the retrieval effectiveness and the computation time complexity when related concepts are treated as unrelated in the phrase-based document similarity calculations.

6. Conclusion

The stem-based VSM that represents documents as vectors in a stem space have been shown to be an effective document representation and retrieval model. Many approaches have been proposed to incorporate phrases or concepts into automatic document retrieval with little success.

In this research, we proposed a new vector space model, the phrase-based VSM, for document retrieval. In the phrase-based VSM, we divided each document into a set of phrases. Each phrase represented a concept in a controlled vocabulary and consisted of several word stems. We derived the similarity between concepts using their relation in a knowledge base, and measured the similarity between two phrases using their stem overlaps and the similarity between the concepts they represented. The similarity between two documents was then defined to be the cosine of the angle between their respective phrase vectors.

Using UMLS as both the controlled vocabulary and the knowledge base to derive the conceptual similarities, we showed from different perspectives, that the retrieval effectiveness of the phrase-based VSM was significantly higher than that of the current gold-standard—the stem-based VSM. Such significant increase of the retrieval effectiveness was achieved without sacrificing too much computation efficiency.

References

[1] Wenlei Mao, Wesley W. Chu, Free-text medical document retrieval via phrase-based vector space model, in: Proceedings of the Annual AMIA Symposium, San Antonio, TX, November 2002, pp. 489–493.
[2] NLM. PubMed overview, 2003. Available from: <http://www.ncbi.nlm.nih.gov/entrez/query/static/overview.html>.

[3] US National Library of Medicine. NLM Gateway, 2003. Available from: <http://gateway.nlm.nih.gov/>.
[4] ASTM International Subcommittee E31.19. E1384-02a standard guide for content and structure of the electronic health record (EHR), 2003. Available from: <http://www.astm.org/>.
[5] Medical Records Institute. Fourth annual MRI survey of electronic health record trends and usage, 2002. Available from: <http://www.medrecinst.com/resources/survey/survey02/index.shtml>.
[6] The US National Library of Medicine and the National Institutes of Health. MEDLINEplus, 2003. Available from: <http://medlineplus.gov/>.
[7] JAMIA, Information for authors, Journal of the American Medical Informatics Association 10 (1) (2003) 110–114.
[8] G. Salton, A. Wang, C.S. Yang, A vector space model for automatic indexing, Communication of the ACM 18 (11) (1975) 613–620.
[9] David D. Lewis, W. Bruce Croft, Term clustering of syntactic phrases, in: Proceedings of the 13th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1990, pp. 385–404.
[10] David A. Hull, Gregory Grefenstette, B. Maximilian Schulze, Eric Gaussier, Hinrich Schütze, Jan O. Pedersen, Xerox TREC-5 site report: routing, filtering, nlp, and Spanish tracks, in: Proceedings of the 5th Text REtrieval Conference (TREC-5), Gaithersburg, MD, November 1996, pp. 167–180.
[11] Avi Arampatzis, Th.P. van der Weide, C.H.A Koster, P. van Bommel, An evaluation of linguistically-motivated indexing schemes, in: Proceedings of the BCSIRSG'2000, 2000.
[12] Chris Buckley, Amit Singhal, Mandar Mitra, Gerard Salton. New retrieval approaches using SMART: TREC 4, in: Proceedings of the 4th Text REtrieval Conference (TREC-4), Gaithersburg, MD, November 1995, pp. 25–28.
[13] Chenxiang Zhai, Xiang Tong, Nataša Milić-Frayling, David A. Evans, Evaluation of syntactic phrase indexing—CLARIT NLP track report, in: Proceedings of the 5th Text REtrieval Conference (TREC-5), Gaithersburg, MD, November 1996, pp. 347–358.
[14] David A. Evans, Chengxiang Zhai, Noun-phrase analysis in unrestricted text for information retrieval, in: Proceedings of 34th Annual Meeting of the Association for Computational Linguistics, Santa Cruz, US, 1996, pp. 17–24.
[15] David B. Johnson, Wesley W. Chu, John D. Dionisio, Ricky K. Taira, Hooshang Kangarloo, Creating and indexing teaching files from free-text patient reports, in: Proceedings of the AMIA Annual Symposium 1999, ~~1999,~~ pp. 814–818.
[16] Joel L. Fagan, Experiments in Automatic Phrase Indexing for Document Retrieval: Comparison of Syntactic and Non-syntactic Methods, PhD thesis, Cornell University, 1988.
[17] Mandar Mitra, Christopher Buckley, Amit Singhal, Claire Cardie, An analysis of statistical and syntactic phrases, in: Proceedings of RIAO'97, 5th International Conference "Recherche d'Information Assistee par Ordinateur", 1997, pp. 200–214.
[18] NLM, UMLS Knowledge Sources, 12th ed., 2001.
[19] Roy Rada, Ellen Bicknell, Ranking documents with a thesaurus, Journal of the American Society for Information Science 40 (5) (1989) 304–310.
[20] NLM, Medical Subject Headings, National Technical Information Service, 1987 (Chapter: Medical subject headings, tree structures).
[21] William R. Hersh, David H. Hickam, T.J. Leone, Words, concepts, or both: optimal indexing unit for automatic information retrieval, in: Mark E. Frisse (Ed.), Proceedings of the 16th Annual Symposium on Computer Applications in Medical Care (SCAMC'92), vol. 16, 1992, pp. 644–648.
[22] Yiming Yang, Christopher G. Chute, Words or concepts: the features of indexing units and their optimal use in information retrieval, in: Proceedings of 17th Annual Symposium on Computer Applications in Medical Care (SCAMC'93), vol. 17, 1993, pp. 685–689.
[23] Christiane Fellbaum (Ed.), WordNet: An Electronic Lexical Database, MIT Press, 1998.
[24] Ellen M. Voorhees, Using WordNet to disambiguate word sense for text retrieval, in: Proceedings of the 16th Annual ACM SIGIR Conference on Research and Development in Information Retrieval, 1993, pp. 171–180.
[25] R. Richardson, A.F. Smeaton, Using WordNet in a knowledge-based approach to information retrieval, Technical Report CA-0395, Dublin City University, Dublin, Ireland, 1995.
[26] Michael Sussna. Text Retrieval using Inference in Semantic Metanetworks, PhD thesis, University of California, San Diego, 1997.
[27] Rada Mihalcea, Dan Moldovan. Semantic indexing using WordNet senses, in: Proceedings of ACL Workshop on IR and NLP, 2000.
[28] Julio Gonzalo, Felisa Verdejo, Irina Chugur, Juan Cigarrán, Indexing with WordNet synsets can improve text retrieval, in: Proceedings of the COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems, 1998, pp. 38–44.
[29] William Hersh, Chris Buckley, T.J. Leone, David Hickam, OHSUMED: an interactive retrieval evaluation and new large test collection for research, in: Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1994, pp. 192–201.
[30] J.B. Lovins, Development of a stemming algorithm, Mechanical Translation and Computational Linguistics 11 (1–2) (1968) 22–31.
[31] M.F. Porter, An algorithm for suffix stripping, Program 14 (3) (1980) 130–137.
[32] Nancy Ide, Jean Véronis, Word sense disambiguation: the state of the art, Computational Linguistics 24 (1) (1998) 1–40.
[33] George A. Miller, Richard Beckwith, Christiane Fellbaum, Derek Gross, Katherine Miller, Introduction to WordNet: an on-line lexical database, in: Five Papers on WordNet, Cognitive Science Laboratory, Princeton University, 1993.
[34] John Lyons, Semantics, Cambridge University Press, 1977.
[35] G. Salton, A new comparison between conventional indexing (MEDLARS) and automatic text processing (SMART), Journal of the American Society for Information Science 23 (2) (1975) 75–84.
[36] Dan Gusfield, Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology, Cambridge University Press, 1997.
[37] Alfred V. Aho, Margaret J. Corasick, Efficient string matching: an aid to bibliographic search, Communications of the ACM 18 (6) (1975) 333–340.

[38] Gerard Salton, Michael J. McGill, The smart and sire experimental retrieval systems, in: Introduction to Modern Information Retrieval [40], pp. 118–156 (Chap. 4).
[39] C.J. van Rijsbergen, Information Retrieval, Butterworth, 1979.
[40] Gerard Salton, Michael J. McGill, Introduction to Modern Information Retrieval, McGraw-Hill Computer Science Series, McGraw-Hill, Inc., 1983.
[41] Chris Buckley, Ellen M. Voorhees, Evaluating evaluation measure stability, in: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2000, pp. 33–40.
[42] Michael Steinbach, George Karypis, Vipin Kumar, A comparison of document clustering techniques, in: Proceedings of the KDD Workshop on Text Mining, 2000.
[43] Ying Zhao, George Karypis, Evaluation of hierarchical clustering algorithms for document datasets, Technical Report 02-022, Department of Computer Science, University of Minnesota, 2002.