US009569505B2

(12) **United States Patent**
Patterson

(10) **Patent No.: US 9,569,505 B2**
(45) **Date of Patent: *Feb. 14, 2017**

(54) **PHRASE-BASED SEARCHING IN AN INFORMATION RETRIEVAL SYSTEM**

(71) Applicant: **GOOGLE INC.**, Mountain View, CA (US)

(72) Inventor: **Anna L. Patterson**, San Jose, CA (US)

(73) Assignee: **Google Inc.**, Mountain View, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **14/713,374**

(22) Filed: **May 15, 2015**

(65) **Prior Publication Data**

US 2015/0248415 A1 Sep. 3, 2015

Related U.S. Application Data

(60) Continuation of application No. 13/919,830, filed on Jun. 17, 2013, now Pat. No. 9,037,573, which is a continuation of application No. 13/309,273, filed on Dec. 1, 2011, now Pat. No. 8,489,628, which is a continuation of application No. 12/717,687, filed on Mar. 4, 2010, now Pat. No. 8,108,412, which is a division of application No. 10/900,012, filed on Jul. 26, 2004, now Pat. No. 7,711,679.

(51) **Int. Cl.**

G06F 7/00	(2006.01)
G06F 17/30	(2006.01)
G06Q 10/10	(2012.01)

(52) **U.S. Cl.**

CPC ***G06F 17/3053*** (2013.01); ***G06F 17/3064*** (2013.01); ***G06F 17/30324*** (2013.01); ***G06F 17/30864*** (2013.01); ***G06Q 10/10*** (2013.01)

(58) **Field of Classification Search**

USPC 707/754, 706, 705, 709, 711, 741, 758,707/722, 723

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,278,980 A 1/1994 Pedersen et al.
5,321,833 A 6/1994 Chang et al.
(Continued)

FOREIGN PATENT DOCUMENTS

CN 1249042 A 3/2000
EP 1622055 A1 2/2006
(Continued)

OTHER PUBLICATIONS

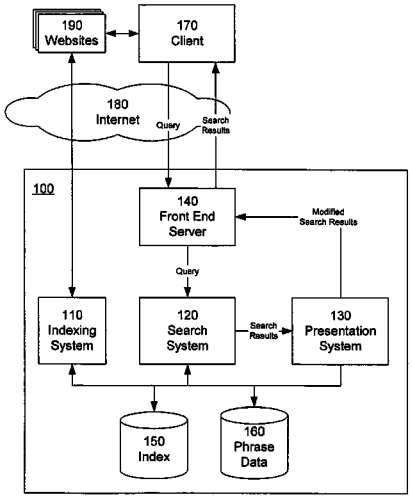
US 7,430,556, 09/2008, Patterson (withdrawn)
(Continued)

Primary Examiner — Hung T Vy
(74) *Attorney, Agent, or Firm* — Brake Hughes Bellermann LLP

(57) **ABSTRACT**

An information retrieval system uses phrases to index, retrieve, organize and describe documents. Phrases are identified that predict the presence of other phrases in documents. Documents are the indexed according to their included phrases. Related phrases and phrase extensions are also identified. Phrases in a query are identified and used to retrieve and rank documents. Phrases are also used to cluster documents in the search results, create document descriptions, and eliminate duplicate documents from the search results, and from the index.

19 Claims, 8 Drawing Sheets



PHRASE-BASED SEARCHING IN AN INFORMATION RETRIEVAL SYSTEM

CROSS REFERENCE TO RELATED APPLICATIONS

This application is a continuation of, and claims priority to, U.S. application Ser. No. 13/919,830, filed on Jun. 17, 2013, now U.S. Pat. No. 9,037,573, entitled "Phrase-Based Personalization of Searches In an Information Retrieval System," which is a continuation of U.S. application Ser. No. 13/309,273, filed on Dec. 1, 2011, entitled "Phrase-Based Detection of Duplicate Documents in an Information Retrieval System", now U.S. Pat. No. 8,489,628, which is a continuation of U.S. application Ser. No. 12/717,687, filed on Mar. 4, 2010, entitled "Phrase-Based Detection of Duplicate Documents in an Information Retrieval System", now U.S. Pat. No. 8,108,412, which is a divisional of U.S. application Ser. No. 10/900,012, filed on Jul. 26, 2004, entitled "Phrase-Based Detection of Duplicate Documents in an Information Retrieval System", now U.S. Pat. No. 7,711,679; and is related to the following applications: U.S. application Ser. No. 10/900,055 filed on Jul. 26, 2004, entitled "Phrase-Based Indexing in an Information Retrieval System", now U.S. Pat. No. 7,536,408; U.S. application Ser. No. 10/900,041 filed on Jul. 26, 2004, entitled "Phrase-Based Searching in an Information Retrieval System", now U.S. Pat. No. 7,599,914; U.S. application Ser. No. 10/900,039 filed on Jul. 26, 2004, entitled "Phrase-Based Personalization of Searches in an Information Retrieval System", now U.S. Pat. No. 7,580,929; U.S. application Ser. No. 10/900,259, filed on Jul. 26, 2004, entitled "Automatic Taxonomy Generation in Search Results Using Phrases", now U.S. Pat. No. 7,426,507; and U.S. application Ser. No. 10/900,075, filed on Jul. 26, 2004, entitled "Phrase-Based Generation of Document Descriptions", now U.S. Pat. No. 7,584,175; all of which are co-owned and incorporated by reference herein.

FIELD OF THE INVENTION

The present invention relates to an information retrieval system for indexing, searching, and classifying documents in a large scale corpus, such as the Internet.

BACKGROUND OF THE INVENTION

Information retrieval systems, generally called search engines, are now an essential tool for finding information in large scale, diverse, and growing corpuses such as the Internet. Generally, search engines create an index that relates documents (or "pages") to the individual words present in each document. A document is retrieved in response to a query containing a number of query terms, typically based on having some number of query terms present in the document. The retrieved documents are then ranked according to other statistical measures, such as frequency of occurrence of the query terms, host domain, link analysis, and the like. The retrieved documents are then presented to the user, typically in their ranked order, and without any further grouping or imposed hierarchy. In some cases, a selected portion of a text of a document is presented to provide the user with a glimpse of the document's content.

Direct "Boolean" matching of query terms has well known limitations, and in particular does not identify documents that do not have the query terms, but have related words. For example, in a typical Boolean system, a search on "Australian Shepherds" would not return documents about other herding dogs such as Border Collies that do not have the exact query terms. Rather, such a system is likely to also retrieve and highly rank documents that are about Australia (and have nothing to do with dogs), and documents about "shepherds" generally.

The problem here is that conventional systems index documents are based on individual terms, rather than on concepts. Concepts are often expressed in phrases, such as "Australian Shepherd," "President of the United States," or "Sundance Film Festival". At best, some prior systems will index documents with respect to a predetermined and very limited set of 'known' phrases, which are typically selected by a human operator. Indexing of phrases is typically avoided because of the perceived computational and memory requirements to identify all possible phrases of say three, four, or five or more words. For example, on the assumption that any five words could constitute a phrase, and a large corpus would have at least 200,000 unique terms, there would approximately 3.2×10^{26} possible phrases, clearly more than any existing system could store in memory or otherwise programmatically manipulate. A further problem is that phrases continually enter and leave the lexicon in terms of their usage, much more frequently than new individual words are invented. New phrases are always being generated, from sources such technology, arts, world events, and law. Other phrases will decline in usage over time.

Some existing information retrieval systems attempt to provide retrieval of concepts by using co-occurrence patterns of individual words. In these systems a search on one word, such as "President" will also retrieve documents that have other words that frequently appear with "President", such as "White" and "House." While this approach may produce search results having documents that are conceptually related at the level of individual words, it does not typically capture topical relationships that inhere between co-occurring phrases.

Accordingly, there is a need for an information retrieval system and methodology that can comprehensively identify phrases in a large scale corpus, index documents according to phrases, search and rank documents in accordance with their phrases, and provide additional clustering and descriptive information about the documents.

SUMMARY OF THE INVENTION

An information retrieval system and methodology uses phrases to index, search, rank, and describe documents in the document collection. The system is adapted to identify phrases that have sufficiently frequent and/or distinguished usage in the document collection to indicate that they are "valid" or "good" phrases. In this manner multiple word phrases, for example phrases of four, five, or more terms, can be identified. This avoids the problem of having to identify and index every possible phrases resulting from the all of the possible sequences of a given number of words.

The system is further adapted to identify phrases that are related to each other, based on a phrase's ability to predict the presence of other phrases in a document. More specifically, a prediction measure is used that relates the actual co-occurrence rate of two phrases to an expected co-occurrence rate of the two phrases. Information gain, as the ratio of actual co-occurrence rate to expected co-occurrence rate, is one such prediction measure. Two phrases are related where the prediction measure exceeds a predetermined threshold. In that case, the second phrase has significant

information gain with respect to the first phrase. Semantically, related phrases will be those that are commonly used to discuss or describe a given topic or concept, such as "President of the United States" and "White House." For a given phrase, the related phrases can be ordered according to their relevance or significance based on their respective prediction measures.

An information retrieval system indexes documents in the document collection by the valid or good phrases. For each phrase, a posting list identifies the documents that contain the phrase. In addition, for a given phrase, a second list, vector, or other structure is used to store data indicating which of the related phrases of the given phrase are also present in each document containing the given phrase. In this manner, the system can readily identify not only which documents contain which phrases in response to a search query, but which documents also contain phrases that are related to query phrases, and thus more likely to be specifically about the topics or concepts expressed in the query phrases.

The use of phrases and related phrases further provides for the creation and use of clusters of related phrases, which represent semantically meaningful groupings of phrases. Clusters are identified from related phrases that have very high prediction measure between all of the phrases in the cluster. Clusters can be used to organize the results of a search, including selecting which documents to include in the search results and their order, as well as eliminating documents from the search results.

The information retrieval system is also adapted to use the phrases when searching for documents in response to a query. The query is processed to identify any phrases that are present in the query, so as to retrieve the associated posting lists for the query phrases, and the related phrase information. In addition, in some instances a user may enter an incomplete phrase in a search query, such as "President of the". Incomplete phrases such as these may be identified and replaced by a phrase extension, such as "President of the United States." This helps ensure that the user's most likely search is in fact executed.

The related phrase information may also be used by the system to identify or select which documents to include in the search result. The related phrase information indicates for a given phrase and a given document, which related phrases of the given phrase are present in the given document. Accordingly, for a query containing two query phrases, the posting list for a first query phrase is processed to identify documents containing the first query phrase, and then the related phrase information is processed to identify which of these documents also contain the second query phrase. These latter documents are then included in the search results. This eliminates the need for the system to then separately process the posting list of the second query phrase, thereby providing faster search times. Of course, this approach may be extended to any number of phrases in a query, yielding in significant computational and timing savings.

The system may be further adapted to use the phrase and related phrase information to rank documents in a set of search results. The related phrase information of a given phrase is preferably stored in a format, such as a bit vector, which expresses the relative significance of each related phrase to the given phrase. For example, a related phrase bit vector has a bit for each related phrase of the given phrase, and the bits are ordered according to the prediction measures (e.g., information gain) for the related phrases. The most significant bit of the related phrase bit vector are associated with the related phrase having the highest prediction measure, and the least significant bit is associated with the related phrase having a lowest prediction measure. In this manner, for a given document and a given phrase, the related phrase information can be used to score the document. The value of the bit vector itself (as a value) may be used as the document score. In this manner documents that contain high order related phrases of a query phrase are more likely to be topically related to the query than those that have low ordered related phrases. The bit vector value may also be used as a component in a more complex scoring function, and additionally may be weighted. The documents can then be ranked according to their document scores.

Phrase information may also be used in an information retrieval system to personalize searches for a user. A user is modeled as a collection of phrases, for example, derived from documents that the user has accessed (e.g., viewed on screen, printed, stored, etc.). More particularly, given a document accessed by user, the related phrases that are present in this document, are included in a user model or profile. During subsequent searches, the phrases in the user model are used to filter the phrases of the search query and to weight the document scores of the retrieved documents.

Phrase information may also be used in an information retrieval system to create a description of a document, for example the documents included in a set of search results. Given a search query, the system identifies the phrases present in the query, along with their related phrases, and their phrase extensions. For a given document, each sentence of the document has a count of how many of the query phrases, related phrases, and phrase extensions are present in the sentence. The sentences of document can be ranked by these counts (individually or in combination), and some number of the top ranking sentences (e.g., five sentences) are selected to form the document description. The document description can then be presented to the user when the document is included in search results, so that the user obtains a better understanding of the document, relative to the query.

A further refinement of this process of generating document descriptions allows the system to provide personalized descriptions, that reflect the interests of the user. As before, a user model stores information identifying related phrases that are of interest to the user. This user model is intersected with a list of phrases related to the query phrases, to identify phrases common to both groups. The common set is then ordered according to the related phrase information. The resulting set of related phrases is then used to rank the sentences of a document according to the number of instances of these related phrases present in each document. A number of sentences having the highest number of common related phrases is selected as the personalized document description.

An information retrieval system may also use the phrase information to identify and eliminate duplicate documents, either while indexing (crawling) the document collection, or when processing a search query. For a given document, each sentence of the document has a count of how many related phrases are present in the sentence. The sentences of document can be ranked by this count, and a number of the top ranking sentences (e.g., five sentences) are selected to form a document description. This description is then stored in association with the document, for example as a string or a hash of the sentences. During indexing, a newly crawled document is processed in the same manner to generate the document description. The new document description can be matched (e.g., hashed) against previous document descrip-

tions, and if a match is found, then the new document is a duplicate. Similarly, during preparation of the results of a search query, the documents in the search result set can be processed to eliminate duplicates.

The present invention has further embodiments in system and software architectures, computer program products and computer implemented methods, and computer generated user interfaces and presentations.

The foregoing are just some of the features of an information retrieval system and methodology based on phrases. Those of skill in the art of information retrieval will appreciate the flexibility of generality of the phrase information allows for a large variety of uses and applications in indexing, document annotation, searching, ranking, and other areas of document analysis and processing.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is block diagram of the software architecture of one embodiment of the present invention.

FIG. **2** illustrates a method of identifying phrases in documents.

FIG. **3** illustrates a document with a phrase window and a secondary window.

FIG. **4** illustrates a method of identifying related phrases.

FIG. **5** illustrates a method of indexing documents for related phrases.

FIG. **6** illustrates a method of retrieving documents based on phrases.

FIG. **7** illustrates operations of the presentation system to present search results.

FIGS. **8***a* and **8***b* illustrate relationships between referencing and referenced documents.

The figures depict a preferred embodiment of the present invention for purposes of illustration only. One skilled in the art will readily recognize from the following discussion that alternative embodiments of the structures and methods illustrated herein may be employed without departing from the principles of the invention described herein.

DETAILED DESCRIPTION OF THE INVENTION

I. System Overview

Referring now to FIG. **1**, there is shown the software architecture of an embodiment of a search system **100** in accordance with one embodiment of present invention. In this embodiment, the system includes a indexing system **110**, a search system **120**, a presentation system **130**, and a front end server **140**.

The indexing system **110** is responsible for identifying phrases in documents, and indexing documents according to their phrases, by accessing various websites **190** and other document collections. The front end server **140** receives queries from a user of a client **170**, and provides those queries to the search system **120**. The search system **120** is responsible for searching for documents relevant to the search query (search results), including identifying any phrases in the search query, and then ranking the documents in the search results using the presence of phrases to influence the ranking order. The search system **120** provides the search results to the presentation system **130**. The presentation system **130** is responsible for modifying the search results including removing near duplicate documents, and generating topical descriptions of documents, and providing the modified search results back to the front end server **140**, which provides the results to the client **170**. The system **100** further includes an index **150** that stores the indexing information pertaining to documents, and a phrase data store **160** that stores phrases, and related statistical information.

In the context of this application, "documents" are understood to be any type of media that can be indexed and retrieved by a search engine, including web documents, images, multimedia files, text documents, PDFs or other image formatted files, and so forth. A document may have one or more pages, partitions, segments or other components, as appropriate to its content and type. Equivalently a document may be referred to as a "page," as commonly used to refer to documents on the Internet. No limitation as to the scope of the invention is implied by the use of the generic term "documents." The search system **100** operates over a large corpus of documents, such as the Internet and World Wide Web, but can likewise be used in more limited collections, such as for the document collections of a library or private enterprises. In either context, it will be appreciated that the documents are typically distributed across many different computer systems and sites. Without loss of generality then, the documents generally, regardless of format or location (e.g., which website or database) will be collectively referred to as a corpus or document collection. Each document has an associated identifier that uniquely identifies the document; the identifier is preferably a URL, but other types of identifiers (e.g., document numbers) may be used as well. In this disclosure, the use of URLs to identify documents is assumed.

II. Indexing System

In one embodiment, the indexing system **110** provides three primary functional operations: 1) identification of phrases and related phrases, 2) indexing of documents with respect to phrases, and 3) generation and maintenance of a phrase-based taxonomy. Those of skill in the art will appreciate that the indexing system **110** will perform other functions as well in support of conventional indexing functions, and thus these other operations are not further described herein. The indexing system **110** operates on an index **150** and data repository **160** of phrase data. These data repositories are further described below.

1. Phrase Identification

The phrase identification operation of the indexing system **110** identifies "good" and "bad" phrases in the document collection that are useful to indexing and searching documents. In one aspect, good phrases are phrases that tend to occur in more than certain percentage of documents in the document collection, and/or are indicated as having a distinguished appearance in such documents, such as delimited by markup tags or other morphological, format, or grammatical markers. Another aspect of good phrases is that they are predictive of other good phrases, and are not merely sequences of words that appear in the lexicon. For example, the phrase "President of the United States" is a phrase that predicts other phrases such as "George Bush" and "Bill Clinton." However, other phrases are not predictive, such as "fell down the stairs" or "top of the morning," "out of the blue," since idioms and colloquisms like these tend to appear with many other different and unrelated phrases. Thus, the phrase identification phase determines which phrases are good phrases and which are bad (i.e., lacking in predictive power).

Referring to now FIG. **2**, the phrase identification process has the following functional stages:

200: Collect possible and good phrases, along with frequency and co-occurrence statistics of the phrases.

202: Classify possible phrases to either good or bad phrases based on frequency statistics.

204: Prune good phrase list based on a predictive measure derived from the co-occurrence statistics.

Each of these stages will now be described in further detail.

The first stage **200** is a process by which the indexing system **110** crawls a set of documents in the document collection, making repeated partitions of the document collection over time. One partition is processed per pass. The number of documents crawled per pass can vary, and is preferably about 1,000,000 per partition. It is preferred that only previously uncrawled documents are processed in each partition, until all documents have been processed, or some other termination criteria is met. In practice, the crawling continues as new documents are being continually added to the document collection. The following steps are taken by the indexing system **110** for each document that is crawled.

Traverse the words of the document with a phrase window length of n, where n is a desired maximum phrase length. The length of the window will typically be at least 2, and preferably 4 or 5 terms (words). Preferably phrases include all words in the phrase window, including what would otherwise be characterized as stop words, such as "a", "the," and so forth. A phrase window may be terminated by an end of line, a paragraph return, a markup tag, or other indicia of a change in content or format.

FIG. **3** illustrates a portion of a document **300** during a traversal, showing the phrase window **302** starting at the word "stock" and extending 5 words to the right. The first word in the window **302** is candidate phrase i, and the each of the sequences i+1, i+2, i+3, i+4, and i+5 is likewise a candidate phrase. Thus, in this example, the candidate phrases are: "stock", "stock dogs", "stock dogs for", "stock dogs for the", "stock dogs for the Basque", and "stock dogs for the Basque shepherds".

In each phrase window **302**, each candidate phrase is checked in turn to determine if it is already present in the good phrase list **208** or the possible phrase list **206**. If the candidate phrase is not present in either the good phrase list **208** or the possible phrase list **206**, then the candidate has already been determined to be "bad" and is skipped.

If the candidate phrase is in the good phrase list **208**, as entry then the index **150** entry for phrase g_j is updated to include the document (e.g., its URL or other document identifier), to indicate that this candidate phrase g_j appears in the current document. An entry in the index **150** for a phrase g_j (or a term) is referred to as the posting list of the phrase g_j. The posting list includes a list of documents d (by their document identifiers, e.g. a document number, or alternatively a URL) in which the phrase occurs.

In addition, the co-occurrence matrix **212** is updated, as further explained below. In the very first pass, the good and bad lists will be empty, and thus, most phrases will tend to be added to the possible phrase list **206**.

If the candidate phrase is not in the good phrase list **208** then it is added to the possible phrase list **206**, unless it is already present therein. Each entry p on the possible phrase list **206** has three associated counts:

P(p): Number of documents on which the possible phrase appears;

S(p): Number of all instances of the possible phrase; and

M(p): Number of interesting instances of the possible phrase. An instance of a possible phrase is "interesting" where the possible phrase is distinguished from neighboring content in the document by grammatical or format markers, for example by being in boldface, or underline, or as anchor text in a hyperlink, or in quotation marks. These (and other) distinguishing appearances are indicated by various HTML markup language tags and grammatical markers. These statistics are maintained for a phrase when it is placed on the good phrase list **208**.

In addition the various lists, a co-occurrence matrix **212** (G) for the good phrases is maintained. The matrix G has a dimension of m×m, where m is the number of good phrases. Each entry G(j, k) in the matrix represents a pair of good phrases (g_j, g_k). The co-occurrence matrix **212** logically (though not necessarily physically) maintains three separate counts for each pair (g_j, g_k) of good phrases with respect to a secondary window **304** that is centered at the current word i, and extends +/−h words. In one embodiment, such as illustrated in FIG. **3**, the secondary window **304** is 30 words. The co-occurrence matrix **212** thus maintains:

R(j,k): Raw Co-occurrence count. The number of times that phrase g_j appears in a secondary window **304** with phrase g_k;

D(j,k): Disjunctive Interesting count. The number of times that either phrase g_j or phrase g_k appears as distinguished text in a secondary window; and

C(j,k): Conjunctive Interesting count: the number of times that both g_j and phrase g_k appear as distinguished text in a secondary window. The use of the conjunctive interesting count is particularly beneficial to avoid the circumstance where a phrase (e.g., a copyright notice) appears frequently in sidebars, footers, or headers, and thus is not actually predictive of other text.

Referring to the example of FIG. **3**, assume that the "stock dogs" is on the good phrase list **208**, as well as the phrases "Australian Shepherd" and "Australian Shepard Club of America". Both of these latter phrases appear within the secondary window **304** around the current phrase "stock dogs". However, the phrase "Australian Shepherd Club of America" appears as anchor text for a hyperlink (indicated by the underline) to website. Thus the raw co-occurrence count for the pair {"stock dogs", "Australian Shepherd"} is incremented, and the raw occurrence count and the disjunctive interesting count for {"stock dogs", "Australian Shepherd Club of America"} are both incremented because the latter appears as distinguished text.

The process of traversing each document with both the sequence window **302** and the secondary window **304**, is repeated for each document in the partition.

Once the documents in the partition have been traversed, the next stage of the indexing operation is to update **202** the good phrase list **208** from the possible phrase list **206**. A possible phrase p on the possible phrase list **206** is moved to the good phrase list **208** if the frequency of appearance of the phrase and the number of documents that the phrase appears in indicates that it has sufficient usage as semantically meaningful phrase.

In one embodiment, this is tested as follows. A possible phrase p is removed from the possible phrase list **206** and placed on the good phrase list **208** if:

a) P(p)>10 and S(p)>20 (the number of documents containing phrase p is more than 10, and the number of occurrences of phrase p is more then 20); or

b) M(p)>5 (the number of interesting instances of phrase p is more than 5).

These thresholds are scaled by the number of documents in the partition; for example if 2,000,000 documents are crawled in a partition, then the thresholds are approximately doubled. Of course, those of skill in the art will appreciate that the specific values of the thresholds, or the logic of testing them, can be varied as desired.

If a phrase p does not qualify for the good phrase list **208**, then it is checked for qualification for being a bad phrase. A phrase p is a bad phrase if:

a) number of documents containing phrase, P(p)<2; and

b) number of interesting instances of phrase, M(p)=0.

These conditions indicate that the phrase is both infrequent, and not used as indicative of significant content and again these thresholds may be scaled per number of documents in the partition.

It should be noted that the good phrase list **208** will naturally include individual words as phrases, in addition to multi-word phrases, as described above. This is because each the first word in the phrase window **302** is always a candidate phrase, and the appropriate instance counts will be accumulated. Thus, the indexing system **110** can automatically index both individual words (i.e., phrases with a single word) and multiple word phrases. The good phrase list **208** will also be considerably shorter than the theoretical maximum based on all possible combinations of m phrases. In typical embodiment, the good phrase list **208** will include about 6.5×10^5 phrases. A list of bad phrases is not necessary to store, as the system need only keep track of possible and good phrases.

By the final pass through the document collection, the list of possible phrases will be relatively short, due to the expected distribution of the use of phrases in a large corpus. Thus, if say by the 10^{th} pass (e.g., 10,000,000 documents), a phrase appears for the very first time, it is very unlikely to be a good phrase at that time. It may be new phrase just coming into usage, and thus during subsequent crawls becomes increasingly common. In that case, its respective counts will increases and may ultimately satisfy the thresholds for being a good phrase.

The third stage of the indexing operation is to prune **204** the good phrase list **208** using a predictive measure derived from the co-occurrence matrix **212**. Without pruning, the good phrase list **208** is likely to include many phrases that while legitimately appearing in the lexicon, themselves do not sufficiently predict the presence of other phrases, or themselves are subsequences of longer phrases. Removing these weak good phrases results in a very robust likely of good phrases. To identify good phrases, a predictive measure is used which expresses the increased likelihood of one phrase appearing in a document given the presence of another phrase. This is done, in one embodiment, as follows:

As noted above, the co-occurrence matrix **212** is an m×m matrix of storing data associated with the good phrases. Each row j in the matrix represents a good phrase g_j and each column k represented a good phrase g_k. For each good phrase g_j, an expected value $E(g_j)$ is computed. The expected value E is the percentage of documents in the collection expected to contain g_j. This is computed, for example, as the ratio of the number of documents containing g_j to the total number T of documents in the collection that have been crawled: P(j)/T.

As noted above, the number of documents containing g_j is updated each time g_j appears in a document. The value for $E(g_j)$ can be updated each time the counts for g_j are incremented, or during this third stage.

Next, for each other good phrase g_k (e.g., the columns of the matrix), it is determined whether g_j predicts g_k. A predictive measure for g_j is determined as follows:

i) compute the expected value $E(g_k)$. The expected co-occurrence rate E(j,k) of g_j and g_k, if they were unrelated phrases is then $E(g_j)*E(g_k)$;

ii) compute the actual co-occurrence rate A(j,k) of g_j and g_k. This is the raw co-occurrence count R(j, k) divided by T, the total number of documents;

iii) g_j is said to predict g_k where the actual co-occurrence rate A(j,k) exceeds the expected co-occurrence rate E(j,k) by a threshold amount.

In one embodiment, the predictive measure is information gain. Thus, a phrase g_j predicts another phrase g_k when the information gain I of g_k in the presence of g_j exceeds a threshold. In one embodiment, this is computed as follows:

$$I(j,k)=A(j,k)/E(j,k)$$

And good phrase g_j predicts good phrase g_k where:

$$I(j,k)>\text{Information Gain threshold.}$$

In one embodiment, the information gain threshold is 1.5, but is preferably between 1.1 and 1.7. Raising the threshold over 1.0 serves to reduce the possibility that two otherwise unrelated phrases co-occur more than randomly predicted.

As noted the computation of information gain is repeated for each column k of the matrix G with respect to a given row j. Once a row is complete, if the information gain for none of the good phrases g_k exceeds the information gain threshold, then this means that phrase g_j does not predict any other good phrase. In that case, g_j is removed from the good phrase list **208**, essentially becoming a bad phrase. Note that the column j for the phrase g_j is not removed, as this phrase itself may be predicted by other good phrases.

This step is concluded when all rows of the co-occurrence matrix **212** have been evaluated.

The final step of this stage is to prune the good phrase list **208** to remove incomplete phrases. An incomplete phrase is a phrase that only predicts its phrase extensions, and which starts at the left most side of the phrase (i.e., the beginning of the phrase). The "phrase extension" of phrase p is a super-sequence that begins with phrase p. For example, the phrase "President of" predicts "President of the United States", "President of Mexico", "President of AT&T", etc. All of these latter phrases are phrase extensions of the phrase "President of" since they begin with "President of" and are super-sequences thereof.

Accordingly, each phrase g_j remaining on the good phrase list **208** will predict some number of other phrases, based on the information gain threshold previously discussed. Now, for each phrase g_j the indexing system **110** performs a string match with each of the phrases g_k that it predicts. The string match tests whether each predicted phrase g_k is a phrase extension of the phrase g_j. If all of the predicted phrases g_k are phrase extensions of phrase g_j, then phrase g_j is incomplete, and is removed from the good phrase list **208**, and added to an incomplete phrase list **216**. Thus, if there is at least one phrase g_k that is not an extension of g_j, then g_j is complete, and maintained in the good phrase list **208**. For example then, "President of the United" is an incomplete phrase because the only other phrase that it predicts is "President of the United States" which is an extension of the phrase.

The incomplete phrase list **216** itself is very useful during actual searching. When a search query is received, it can be compared against the incomplete phase list **216**. If the query (or a portion thereof) matches an entry in the list, then the