SEARCH ENGINE FOR TEXTUAL CONTENT AND NON-TEXTUAL CONTENT

(57) Abstract: A search engine system that can match a search request to not only a specific content item (e.g., video file), but also to a single component of a content item. For instance, using a video content item as an example, the search engine system can match a specific search request to not only a specific video within a collection of videos, but also to a single moment within a video, a video segment, and a group of videos.

FIG. 5
TECHNICAL FIELD

This disclosure relates generally to search engine systems and apparatus, methods, computer programs and computer program products therefore.

BACKGROUND

Search engine systems (also referred to simply as "search engines") are systems that assist a user in finding information that the user wishes to obtain. Today, many of the most common search engines are Internet based and operate in a client-server environment that includes a client system (e.g., a web page displayed by a computer) that enables a user to submit to a search engine a search request. The search request typically includes one or more query terms, and each query term typically includes a word or a phrase. The search engine, in response to receiving a search request, typically compares the query terms (a.k.a. "keywords") against an index created from a multitude of content items, such as text files (e.g., ascii files), image files (e.g., jpg files, gif files), video files (e.g., .mpg files, .swf files, .avi files), web pages, and other content items, and, based on the comparison, returns an indication of the most relevant content items. The classic example of a search engine is an Internet search engine that uses user-provided keywords to find relevant web pages and returns a list of hyperlinks to the most relevant web pages.

For instance, a user may submit a search request to a search engine located at www.google.com (as retrieved on 2 May 2013). In response, the search engine will present a number of results, such as a list of web pages that match the query terms included in the search request, with the most relevant results often being displayed at the top of the returned web page. Similarly, a user may submit a search request to a search engine located at www.youtube.com (as retrieved on 2 May 2013) and receive a list of matching videos.

As the amount of digital data increases, search engine systems are being deployed not only for Internet search, but also for proprietary, personal, or special-purpose databases, such as personal archives, user generated content sites, proprietary data stores, workplace databases,
and others. For example, personal computers may host a search engine to find content items stored anywhere on the hard-drive of the computer or in special-purpose archives (e.g., personal music or video collection) stored on the hard-drive.

[05] Given this tremendous growth in the amount of digital data that is accessible to a user, particularly "non-textual" digital data, which we define as digital data that includes non-text data, such as, for example, video data, audio data, image data, etc., there remains a need to improve upon the existing search engine systems.

SUMMARY

[06] The inventors have discovered that an improved search engine system is a system that can match a search request to not only a specific content item (e.g., video file), but also to a single component of a content item. For instance, using a video content item as an example, the inventors have discovered that it would be advantageous to implement a search engine system so that it can match a specific search request to not only a specific video within a collection of videos, but also to a single moment within a video (e.g., a video frame), a time span within a video (e.g., a video segment) and a group of videos in the video collection. Described herein are implementations of such a search engine system.

[07] For example, in one aspect of this disclosure, there is provided a method performed by a search engine system (SES). In one embodiment, the method includes receiving, at the SES, a search request transmitted by a client device. The search request includes one or more query terms. The method also includes the SES determining i) a query vector based on the one or more query terms, ii) a first segment vector for a first segment of a first non-textual content item, and iii) a second segment vector for a second segment of the first non-textual content item. The method also includes the SES calculating i) a first segment search score based on the result of a comparison of the first segment vector to the query vector, and ii) a second segment search score based on a result of a comparison of the second segment vector to the query vector.

[08] In one embodiment, the first segment is associated with a first set of tags, the second segment is associated with a second set of tags, the first set of tags includes the first tag, the second set of tags includes the second tag. In such an embodiment, the method also includes determining a first set of tag vectors based on the first set of tags and the one or more query
terms; and determining a second set of tag vectors based on the second set of tags and the one or more query terms. In this embodiment, the step of determining the first segment vector comprises summing the first set of tag vectors, and the step of determining the second segment vector comprises summing the second set of tag vectors.

[09] In one embodiment, the method also includes the steps of: determining, by the SES, a first item vector for the first non-textual content item; determining, by the SES, a second item vector for a second non-textual content item; determining, by the SES, a first item search score, wherein the first item search score is based on a comparison of the first item vector to the query vector; determining, by the SES, a second item search score, wherein the second item search score is based on a comparison of the second item vector to the query vector; and selecting one or more of: the first segment, the second segment, the first non-textual content item, and the second non-textual content item based on the first segment search score, second segment search score, first item search score, and second item search score.

[010] In one embodiment, the method also includes the step of transmitting an ordered set of two or more search results based on the search request, wherein the ordered set of search results includes a first search result that comprises information identifying the first segment, wherein the position of the first search result within the ordered set of search results is determined based on the first segment search score and a search score associated with each other search result included in the ordered set of search results.

[011] In another embodiment, the method performed by the SES includes determining, by the SES, a first tag vector based on the one or more query terms and a first tag, wherein the first tag is linked with a first feature located in a first segment of non-textual content item; determining, by the SES, a second tag vector based on the one or more query terms and a second tag, wherein the second tag is linked with a second feature located in a second segment of the non-textual content item; calculating, by the SES, a first tag search score based on the result of a comparison of the first tag vector to the query vector; and calculating, by the SES, a second tag search score based on the result of a comparison of the second tag vector to the query vector.

[012] In one embodiment, the first set of tag vectors comprises a first weighted tag vector, and determining the first set of tag vectors comprises determining a first initial tag vector for the first tag and multiplying the first initial tag vector with a featurescore associated with a feature
type of the first tag, thereby producing the first weighted tag vector. The feature type may be
one of image, audio, video, and text.

[013] The search request may include a search type indicator. The search type indicator
may indicate that the user is requesting, a the least, a tag search.

[014] In another aspect, search engine system (SES) is provided. In one embodiment, the
SES comprises a data storage system. The SES also includes a data processing system. The data
storage system includes instructions executable by the data processing system whereby the SES
is operative to: determine a query vector based on query terms included in a search
request; determine a first segment vector for a first segment of a first non-textual content
item; determine a second segment vector for a second segment of the first non-textual content
item; calculate a first segment search score based on the result of a comparison of the first
segment vector to the query vector; and calculate a second segment search score based on a result
of a comparison of the second segment vector to the query vector.

[015] In one embodiments, the SES is operative to calculate the first segment search score
by, at least, calculating: \((VQ \cdot VS1) / (||VQ|| ||VS1||)\), where VQ is the query vector, and VS1 is
the first segment vector, and calculate the second segment search score by, at least, calculating:
\((VQ \cdot VS2) / (||VQ|| ||VS2||)\), where VS2 is the second segment vector.

[016] In one embodiments, the SES is also operative to determine a first item vector for the
first non-textual content item; determine a second item vector for a second non-textual content
item; determine a first item search score, wherein the first item search score is based on a
comparison of the first item vector to the query vector; and determine a second item search score,
wherein the second item search score is based on a comparison of the second item vector to the
query vector. The SES may further be operative to select one or more of: the first segment, the
second segment, the first non-textual content item, and the second non-textual content item based
on the first segment search score, second segment search score, first item search score, and
second item search score.

[017] In another embodiment, the SES is operative to: determine a query vector based on
query terms included in a search request; determine a first tag vector based on the one or more
query terms and a first tag, wherein the first tag is linked with a first feature located in a first
segment of a non-textual content item; determine a second tag vector based on the one or more query terms and a second tag, wherein the second tag is linked with a second feature located in a second segment of the non-textual content item; calculate a first tag search score based on the result of a comparison of the first tag vector to the query vector; and calculate a second tag search score based on the result of a comparison of the second tag vector to the query vector.

[018] In another aspect, there is provided a search engine apparatus. In one embodiment, the search engine apparatus comprises a receiver unit configured to receive a search request transmitted by a client device. The search request includes one or more query terms. The search engine apparatus also includes a vector determining unit. The vector determining unit is configured to: determine a query vector based on the one or more query terms; determine a first segment vector for a first segment of a first non-textual content item; and determine a second segment vector for a second segment of the first non-textual content item. The search engine apparatus also includes a search score calculating unit. The search score calculating unit is configured to: calculate a first segment search score based on the result of a comparison of the first segment vector to the query vector; and calculate a second segment search score based on a result of a comparison of the second segment vector to the query vector.

[019] In another embodiments, the vector determining unit is configured to: determine a query vector based on the one or more query terms; determine a first tag vector based on the one or more query terms and a first tag, wherein the first tag is linked with a first feature located in the first segment of the non-textual content item; and determine a second tag vector based on the one or more query terms and a second tag, wherein the second tag is linked with a second feature located in the second segment of the non-textual content item. In this embodiments, the search score calculating unit is configured to: calculate a first tag search score based on the result of a comparison of the first tag vector to the query vector; and calculate a second tag search score based on the result of a comparison of the second tag vector to the query vector.

[020] In another aspect, a computer program product is provided. The computer program product includes a non-transitory computer readable medium storing computer instructions for searching content.

[021] In one embodiments, the computer instructions include: instructions for determining a query vector based on query terms included in a search request; instructions for determining a
first segment vector for a first segment of a first non-textual content item; instructions for determining a second segment vector for a second segment of the first non-textual content item; instructions for calculating a first segment search score based on the result of a comparison of the first segment vector to the query vector; and instructions for calculating a second segment search score based on a result of a comparison of the second segment vector to the query vector.

[022] In another embodiment, the computer instructions include: instructions for determining a query vector based on query terms included in a search request; instructions for determining a first tag vector based on the one or more query terms and a first tag, wherein the first tag is linked with a first feature located in the first segment of the non-textual content item; instructions for determining a second tag vector based on the one or more query terms and a second tag, wherein the second tag is linked with a second feature located in the second segment of the non-textual content item; instructions for calculating a first tag search score based on the result of a comparison of the first tag vector to the query vector; and instructions for calculating a second tag search score based on the result of a comparison of the second tag vector to the query vector.

[023] In another aspect, a computer program is provided. The computer program includes computer readable instructions.

[024] In one embodiment, the computer readable instructions are configured such that when run on a search engine system, the instructions cause the search engine system to: determine a query vector based on query terms included in a received search request; determine a first segment vector for a first segment of a first non-textual content item; determine a second segment vector for a second segment of the first non-textual content item; calculate a first segment search score based on the result of a comparison of the first segment vector to the query vector; and calculate a second segment search score based on a result of a comparison of the second segment vector to the query vector.

[025] In another embodiments, the computer readable instructions are configured such that when run on a search engine system, the instructions cause the search engine system to: determine a query vector based on query terms included in a received search request; determine a first tag vector based on the one or more query terms and a first tag, wherein the first tag is linked with a first feature located in the first segment of the non-textual content item; determine a
second tag vector based on the one or more query terms and a second tag, wherein the second tag
is linked with a second feature located in the second segment of the non-textual content
item; calculate a first tag search score based on the result of a comparison of the first tag vector to
the query vector; and calculate a second tag search score based on the result of a comparison of
the second tag vector to the query vector.

[026] The above and other aspects and embodiments are further described herein.

BRIEF DESCRIPTION OF THE DRAWINGS

[027] The accompanying drawings, which are incorporated herein and form part of the
specification, illustrate various embodiments.

[028] FIG. 1 is an illustration of a search engine system in accordance one embodiment.

[029] FIG. 2 illustrates a search engine server system according to one embodiment.

[030] FIG. 3 illustrates a hierarchical relationship between videos, video segments, and
tags.

[031] FIG. 4 is an illustration of a vector space in accordance with exemplary
embodiments.

[032] FIG. 5 is a flow chart illustrating a search process in accordance with exemplary
embodiments.

[033] FIG. 6 is a flow chart illustrating a search process in accordance with exemplary
embodiments.

[034] FIG. 7 is a flow chart illustrating a search process in accordance with exemplary
embodiments.

[035] FIG. 8 is an illustration of a relational database in accordance with exemplary
embodiments.

[036] FIG. 9 is a flow chart illustrating a search process in accordance with exemplary
embodiments.

[037] FIG. 10 is a block diagram of a search engine system in accordance with
exemplary embodiments.
[038] FIG. 11 is a block diagram of a search engine apparatus in accordance with exemplary embodiments.

DETAILED DESCRIPTION

[039] Existing search engine techniques do not provide functionality to match a specific search request with a single moment within a video, a time span within a video, and/or a collection of videos. For example, as is the case with the YouTubeTM website, a user who submits a search request is not presented with a moment or segment from a video as a search result. The disclosed system, method, apparatus, computer program, and computer program product overcome these, and other, deficiencies of existing search engines.

[040] When searching for non-textual content items (i.e., content items that include non-text data, such as, for example, video files), there are numerous types of results that may be of interest to the user. For example, a video stored in a video file can be considered, conceptually, as a series of very short moments, which together make up the video as a whole. As such, an optimal search result for a user's query can be one or more of: a specific moment in a video (e.g., a video frame), a segment of a video, and a tag associated with certain features of the video (e.g., images included in the video data, or sounds, text, and meta-data found in the video file that contains the video data). As such, embodiments of the present disclosure are directed to search techniques that enable a user to search for specific tags, moments, and segments within a video, as well entire videos or collections of videos.

[041] FIG. 1 illustrates a system 100 according to an embodiment. According to the illustrated embodiment, system 100 is a distributed, client-server system that includes a group of client devices 101a-c (e.g., smartphone, laptop computer, tablet device, desktop computer), each of which may execute a client computer program 103 (e.g., a web browser or app), connected to a search engine system (SES) 120 via a network 110, which may be a public network (e.g., the Internet), a private network, or some combination of the two. Network 110 may correspond to any type or combination of communication networks, including local-area networks, wide-area networks, wireless networks, cellular networks, etc. Without loss of generality, network 110 may be described herein as the Internet, but there are embodiments where network 110 is, for example, a local area network (LAN), such as a LAN used by a family in the family's home or a corporate network used by a company. While system 100 is illustrated in FIG. 1 as being a
distributed, client-server system, in other embodiments system 100 need not be distributed. For example, in one embodiment, system 100 consists of a single machine 269 (e.g., personal computer) (see FIG. 2A) that has local software 270 stored thereon, where the local software 270 implements the functionality of the components 122, 124, and 126 of SES 120(discussed below) as well as the functionality of computer program 103.

Computer program 103 is operable to cause client devices 101 to transmit search requests to the SES 120 via network 110. In one embodiment, client devices 101 may transmit the search request in accordance with one or more communication protocols. For instance, in some embodiments, a client device 101 may include a search request in a hypertext transfer protocol (HTTP) message(e.g., an HTTP GET request) and may transmit the search request to SES 120 by transmitting to the SES 120 the HTTP message over network 110.

As further illustrated in FIG. 1, in one embodiment, SES 120 includes the following functional components: a query processor (QP) 122, an indexer 124, and a vector generator 126. In some embodiments, SES 120 may be a distributed system. For example, as illustrated in FIG. 2, SES 120 may include a separate specific machine (e.g., a server computer) for each functional component. In the example shown in FIG. 2, SES 120 according to one embodiment includes three machines 222, 224, and 226, where: i) machine 222 performs the query processor functionality by, for example, executing query processor computer code 223 stored on a non-transitory computer readable medium readable by machine 222; ii) machine 224 performs the indexer functionality by, for example, executing indexer computer code 225 stored on a non-transitory computer readable medium readable by machine 224; and iii) machine 226 performs the vector generator functionality by, for example, executing vector generator computer code 227 stored on a non-transitory computer readable medium readable by machine 226. Machines 222, 224 and 226 may be co-located in the same facility or may be located in separate facilities that are geographically separated. In some embodiments, two or more of the functional components of SES 120 may be performed by a single machine. For example, a single machine (e.g., machine 222) may perform the indexer, query processor, and vector generator functionality by, for example, executing query processor computer code 223, indexer computer code 225, and vector generator computer code 227.
Indexer 124 may be configured, for example, to crawl a collection of stored content items 190 (which may be a distributed collection) and create a corresponding set of one or more tags 192 for each crawled content item and a tag index 191. For the sake of simplicity and brevity, we shall assume that the collection of content items 190 consists of a collection of videos. Referring now to FIG. 3, FIG. 3 shows two videos (video 304 and video 306) that are included in collection 190. As further shown, each video is stored in a common directory 302 and each video includes one or more segments (as an example video 304 includes segments S1 and S2; video 306 includes segments S3 and S4). In the example shown, segment S1 includes an image a Lego robot and segment S2 indicates that the video 304 is over by displaying "The End" while music plays.

Indexer 124 may be configured to analyze each segment of each video to create, for each analyzed segment, a set of one or more tags. Such created tags are added to tag set 191. A tag may simply be a word or phrase that represents a feature that indexer 124 found in the segment of the video. For instance, the indexer 124 may be configured to recognize images and text included in a video segment as well as convert the audio of the video segment to text.

Thus, for example, if indexer 124 recognizes an image of a robot in a segment of a video, then indexer 124 may create a tag for this image feature of the segment. The tag may include the word "robot" as well as a type identifier that identifies the type of the feature. In this example, the tag may consist of the following tuple: ["robot", Image]. Likewise, for example, if indexer 124 recognizes that the audio portion of a video segment contains the word "robot" because, for example, a person in the video said "robot," then indexer 124 may create a tag for this audio feature of the segment. The tag may include the word "robot" as well as a type identifier that identifies the type of the feature. In this example, the tag may consist of the following tuple: ["robot", Audio]. Indexer 124 may also create tags from meta-data associated with a video. For example, if the title of the video is "robots from mars" then indexer may create the following tag: ["robots from mars", title meta-data].

Accordingly, after the indexing process, each segment of each video may have an associated set of tags. That is, a set of one or more tags may be linked to a video segment. As illustrated in FIG. 3, tags T1-T3 are linked to segment S1; tags T4-T5 are linked to segment S2, tags T6-T7 are linked to segment S3; and tag T8 is linked to segment S4. In this example, tag T1
indicates that an image of a Legorobot was found in segment SI, tag T2 indicates that the text "I am Lego Robot" is displayed during at least a portion of segment SI, and tag T3 indicates the text of what the robot was saying during segment SI. Similarly, T4 could indicate that the words "The End" are displayed in segment S2 while T5 can provide an indication of the music played during segment S2.

In response to receiving a search request, query processor 122 may access tag set 192 to select and retrieve tags included therein for use in determining videos that match query terms included in the received search request. Query processor 122 may then request vector generator 126 to generate vectors, such as query vector and one or more tag, segment, and/or video vectors based on the query terms included in the search request and the selected tags. Query processor 122, in one embodiment, uses the query vector and the tag/segment/video vectors to determine tags/segments/videos that match the search request. Query processor 122 may determine whether a tag matches the search request by comparing the query vector with the tag vector for the tag. After determining the tag, segments, and/or videos that match the search request, query processor 122 returns to the requesting client a search result (e.g., a web page) having a set of search result hyperlinks where each search result hyperlink identifies tag, segment, video, or collection of videos.

Referring now to FIG. 4, FIG. 4 illustrates example vectors that may be generated by vector generator 126 in response to query processor 122 receiving a search request. In this example, the search request includes only two query terms (q1 and q2), thus all the generated vectors lie on the same plane. As illustrated, there is a query vector (VQ) 402, a first tag vector (VT1) 404 corresponding to tag T1, a second tag vector (VT2) 406 corresponding to tag T2, and a video vector (W) 408 corresponding to video 304.

In one embodiment, VQ = (Wq1,Wq2), where Wq1 is a weight value assigned to the first query term of the search request and Wq2 is a weight value assigned to the second query term of the search request. In the example shown in FIG. 4, Wq1 = Wq2 = 1.

Also, in one embodiment VT1 = (Wql-tl,Wq2-tl), where Wql-tl is a weight value which may be a function of the number of timesql appears in T1 and Wq2-tl is a weight value which may be a function of the number of timesql2 appears in T1. Likewise, VT2 = (Wql-t2,Wq2-t2), where Wql-t2 is a weight value which may be a function of the number of timesql
appears in T2 and Wq2-t2 is a weight value which may be a function of
the number of timesq2 appears in T2. In some embodiments, VT1 = fs-tl * (Wql-tl, Wq2-tl) and VT2 = fs-t2 * (Wql-t2, Wq2-t2), where fs-tl is a feature score for tag T1 and fs-t2 is a feature score for tag T2. The feature score, in some embodiments, is a value assigned to a feature type. For example, as discussed above, a tag may include an identifier that identifies the type of the feature with which the tag is associated. Each such feature type may have a corresponding feature score. For instance, the feature type of "image" may have a feature score of 1.5, whereas the feature score for the feature type of "audio" may have a feature score of .3. Thus, using these features scores as an example, if we assume that T1 = ["robot", Image], then VT1 = 1.5 * (Wql-tl, Wq2-tl). In some embodiments, as shown in FIG. 4, the video vector W = VT1 + VT2.

[052] Once the query vector and tag vectors are determined (e.g., generated, calculated, or obtained), query processor 122 can compare the query vector with the tag vectors to determine a search score for each tag vector, as discussed more fully below. Similar vector constructions and comparisons may be performed at the segment or video level. As such, in some embodiments, all tags, segments and videos in the corpus can be consistently scored and compared to one another. It is also possible to search for only one type of search result, by only comparing the final search score for tags, segments or videos. This scheme also allows searching for some but not all types of search result (for instance videos and segments, but not tags).

[053] More generically, according to some embodiments, a search request (a.k.a., "query (Q)") can be comprised of a set of query terms q1, q2, ... qn, and the query can represented by a query vector, VQ, where VQ = (Wql, Wq2, ..., Wqn), and Wqxn is the weight for the query term qxn in the query Q.

[054] Each tag Tx included in tag set 192 can be represented by a vector, VTx, where, in some embodiments: VTx = fs-tx * (Wql-tx, Wq2-tx, ..., Wqn-tx), where Wqnx is the weight for the query term qnx in tag Tx. In one embodiments, VTx = fs-tx * (Wql-tx, Wq2-tx, ..., Wqn-tx), where fs-tx is a feature score for tag Tx.

[055] Each video included in collection 190 can be represented by a video vector W, where W = VT1 + VT2 + ... + VTn, and where VT1 ... VTn are the tag vectors that represent the tags that are linked to the video. Similarly, each video segment of a video can be represented by
a segment vector VS, where \( V_S = VT_1 + VT_2 + \ldots + VT_m \), and where \( VT_1 \ldots VT_m \) are the tag vectors that represent the tags that are linked to the segment.

According to some embodiments, query processor 122 determines a search score for a tag, segment or video by determining a cosine similarity value between the query vector generated based on the query and the tag, segment, or video vectors, respectively. This can yield, for instance, a real number between 0 and 1, where a higher score signifies a closer match to the search query. For example, a search score, \( RT_x \), for a tag \( T_x \) can be determined according to:

\[
RT_x = (V_Q \cdot VT_x) / (\|V_Q\| \|VT_x\|)
\]

where \( (V_Q \cdot VT_x) / (\|V_Q\| \|VT_x\|) \) is the cosine similarity between query vector \( V_Q \) and tag vector \( VT_x \). The value of \( RT_x \) may be some or all of the search score for a tag. Similarly, a score, \( RS_x \), for a segment \( S_x \) can be determined according to:

\[
RS_x = (V_Q \cdot VS_x) / (\|V_Q\| \|VS_x\|)
\]

where \( (V_Q \cdot VS_x) / (\|V_Q\| \|VS_x\|) \) is the cosine similarity between query vector \( V_Q \) and segment vector \( VS_x \). The value of \( RS_x \) may be some or all of the search score for a segment. Finally, a score, \( RV \), for a video, \( V \), can be determined according to:

\[
RV = (V_Q \cdot W) / (\|V_Q\| \|W\|)
\]

where \( (V_Q \cdot W) / (\|V_Q\| \|W\|) \) is the cosine similarity between query vector \( V_Q \) and video vector \( W \). The value of \( RV \) may be some or all of the search score for a video.

Referring now to FIG. 5, FIG. 5 is a flow chart illustrating a process 500, according to an example use case, which process is performed by SES 120. Process 500 may begin with step 510, where SES 120 receives a search request that includes one or more query terms. In this example use case, the SES 120 receives the search request from client device 101a. The search request, in addition to including query terms, may also indicate the type of content items that a user wishes to search. For instance, the search request may include information indicating the user wants SES 120 to search only for videos. In this example use case, the search request indicates that tags may be included in the search result. As an example, the search request may be an text string of the form:

"?query_terms=term1,term2,term3\&result_type=tags,segments,videos". In some embodiments,
computer program 103 generates the search request and causes client device 101 to transmit the generated search request. Computer program 103 may cause the generation of the search request in response to a user clicking a submit button on a search form. FIG. 12 illustrates an example search form 1200 that may be displayed with the help of the computer program 103. As shown, the example search form enables a user to enter into an input field 1298 of form 1200 one or more query terms. Also, form 1200 includes check boxes 1299 that enables the user to specify the result types. When the user clicks on (or otherwise activates submit query button 1269), computer program 103, in response, causes the generation of a search request, such as the one illustrated above, and causes the client 101 on which it is executing to provide the search request to network 110, which will route the search request to SES 120.

[058] In step 520, SES 120 determines a query vector (QV) based on the query terms. As used herein, determining can mean directly determining, calculating, generating, retrieving, receiving, and/or obtaining from either a local or remote source. According to some embodiments, determining the query vector may include utilizing one or more weights associated with each of the query terms. For example, some terms may be considered more important, thereby effecting the size and direction of the query vector. The query vector may be determined as described above. In step 530, SES 120 determines a first tag vector (VT1) based on the query terms and a first tag (T1). In step 540, SES 120 determines a second tag vector (VT2) based on the query terms and a second tag (T2). In step 550, a first tag search score is determined based on a comparison of the query vector and the first tag vector. The comparison may be, for example, a cosine comparison such as discussed above. In step 560, a second tag search score is determined based on a comparison of the query vector and the second tag vector.

[059] In step 570, SES 120 generates a search result for the received search request. The generated search result, in one embodiment, includes a list of items (e.g., tags, segments, videos) that match the search request. For example, the search result may be a mark-up language document (e.g., an XML document, an HTML document, etc.) that includes a set of hyperlinks where each hyperlink identifies an item that matches the search request. In generating the search result, SES 120 determines whether the first tag should be identified in the search result as matching the search request. This determination is based on the search score for the first tag. For example, in some embodiments, if SES 120 determines that the search score for the first tag
exceeds a threshold value, then SES 120 includes the first tag in the search result (e.g., includes in the markup language document a hyperlink that points to the tag). Likewise, SES 120 determines whether the second tag should be included in the search result. Additionally, depending on the parameter of the search request, SES 120 may determine whether to add to the search result a segment, a video, a collection of videos, etc. In step 580, SES 120 transmits the search result to client 101a. In some embodiments steps 570 and 580 are optional.

[060] Referring now to FIG. 6, FIG. 6 is a flow chart illustrating a process 600, according to another example use case, which process is performed by SES 120. Process 600 may begin with step 610, where SES 120 receives a search request that includes one or more query terms. In this example use case, the SES 120 receives from client device 101a an HTTP message (e.g., GET request) that includes a search request. As discussed above, the search request may indicate the type of content items that a user wishes to search. In this example use case, the search request indicates that segments may be included in the search result.

[061] In step 620, SES 120 determines a query vector (QV) based on the query terms. The query vector may be determined as described above. In step 630, SES 120 determines a first segment vector (VSI) based on the query terms and a first set of tags linked with a first segment of a non-textual content item. In step 640, SES 120 determines a second segment vector (VS2) based on the query terms and a second set of tags linked with a second segment of the non-textual content item. In step 650, a first segment search score is determined based on a comparison of the query vector and the first segment vector. The comparison may be, for example, a cosine comparison such as discussed above. In step 660, a second segment search score is determined based on a comparison of the query vector and the second segment vector.

[062] In step 670, SES 120 generates a search result for the received search request. The generated search result, in one embodiment, includes a list of items (e.g., tags, segments, videos) that match the search request. In generating the search result, SES 120 determines whether the first segment should be identified in the search result as matching the search request. This determination is based on the search score for the first segment. For example, in some embodiments, if SES 120 determines that the search score for the first segment exceeds a threshold value, then SES 120 includes the first segment in the search result (e.g., includes in the markup language document a hyperlink that points to the first segment). Likewise, SES 120
determines whether the second segment should be included in the search result. Additionally, depending on the parameter of the search request, SES 120 may determine whether to add to the search result a tag, a video, a collection of videos, etc. In step 680, SES 120 transmits the search result to client 101a.

[063] Referring now to FIG. 7, FIG. 7 is a flow chart illustrating a process 700, according to another example use case, which process is performed by SES 120. Process 700 may begin with step 710, where SES 120 receives a search request that includes one or more query terms. In this example use case, the SES 120 receives from client device 101a an HTTP message (e.g., GET request) that includes a search request. As discussed above, the search request may indicate the type of content items that a user wishes to search. In this example use case, the search request indicates that videos may be included in the search result.

[064] In step 720, SES 120 determines a query vector (QV) based on the query terms. The query vector may be determined as described above. In step 730, SES 120 determines a first video vector (V1) based on the query terms and a first set of tags linked with a first video. In step 740, SES 120 determines a second video vector (V2) based on the query terms and a second set of tags linked with a second video. In step 750, a first video search score is determined based on a comparison of the query vector and the first video vector. The comparison may be, for example, a cosine comparison such as discussed above. In step 760, a second video search score is determined based on a comparison of the query vector and the second video vector.

[065] In step 770, SES 120 generates a search result for the received search request. The generated search result, in one embodiment, includes a list of items (e.g., tags, segments, videos) that match the search request. In generating the search result, SES 120 determines whether the first video should be identified in the search result as matching the search request. This determination is based on the search score for the first video. For example, in some embodiments, if SES 120 determines that the search score for the first video exceeds a threshold value, then SES 120 includes the first video in the search result (e.g., includes in the markup language document a hyperlink that points to the first video). Likewise, SES 120 determines whether the second video should be included in the search result. Additionally, depending on the parameter of the search request, SES 120 may determine whether to add to the search result a
tag, a segment, a collection of videos, etc. In step 780, SES 120 transmits the search result to client 101a.

[066] According to some embodiments, the processes described above may be performed by a searching apparatus. The apparatus may include, for instance, a number of hardware units, each adapted to perform one or more of the above steps. For example, a searching apparatus could include a receiving unit configured to receive, from a client device such as client devices IOla-c, a search request that includes one or more query terms. The apparatus may also include one or more determining units configured to determine query, tag, segment, and/or video vectors as described above in connection with processes 500, 600, and 700. The determining units may also be configured to determine a search score and identify one or more media elements to return to the client device. In certain aspects, the results may be transmitted by a transmission unit.

[067] Referring now to FIG. 8, FIG. 8 illustrates tag set 192 according to some embodiments, which may or may not be used in one of the embodiments previously mentioned in conjunction with Figs 1-7, but also Figs 9-11. As illustrated, tag set 192 organizes tag, segment, and video information and relationships for SES 120. In the example shown, each video VI - VN is linked with one or more segments. For instance, video VI is linked with segments S1-S4. Similarly, tag set 192 can provide the relationship between each of the segments and their underlying tags. For instance, the first segment S1 of video VI is linked with tags T1 - TN.

[068] Referring now to FIG. 9, FIG. 9 is a flow chart illustrating a hierarchical search process 900, according to some embodiments, performed by SES 120. As illustrated in FIG. 9, hierarchical search process 900 may be performed in different ways depending on, for instance, a search typeselected by a user. For example, a user may select to perform a "tag," a "segment," or a "full" search. For instance, referring to FIG. 12, the user can perform a full search by checking the "All" checkbox 1277. Similarly, a user may select to perform only a tag search by only checking the "Tags" checkbox 1276, and the user may select to perform a tag and segment search by checking checkboxes 1276 and 1279. In short, the user can check one or more of checkboxes 1299. In some embodiments, in the case of a full search, SES 120 performs a search based on all of the available information levels. For instance, in the non-limiting example of
process 900, a full search may include determining a search score for tags, segments, videos, and video collections. Alternatively, in the non-limiting example of process 900, the search engine may only determine a search score for segments and videos. In some embodiments, the user may receive different results based on what type of search is selected.

[069] Before the process 900 begins, as a first step, a user formulates a search request by entering query terms into client 101, e.g. via a virtual or physical keyboard displayed on a display of the client 101 or by voice commands. For instance, the user may enter multiple query terms into a text entry box of a webpage displayed by a web browser (see FIG. 12 for an exemplary page). Another example is via an application running on the client 101 and displaying a GUI on a display of the client device. The application can in one such embodiment communicate with SES 120 via HTTP messages or via an API (Application Programming Interface) for the search service provided by SES 120. For illustration, we will assume the user enters the query terms "Lego Robot." In a second step before the process 900, which may also be a portion of the first step, or performed as a separate and independent action, the user determines what type of search he or she would like to perform. After performing the two steps, the user causes client 101 to provide to search engine server system 120 a search request that includes the query terms entered by the user and a search type indicator (e.g., one or more identifiers that identify whether SES 120 should perform a full or limited search).

[070] In step 902 of process 900, which occurs after the user has entered the query terms, selected a search type (optional), and submitted the information, SES 120 receives a search request (or "query") including the entered query terms and search type indicator identifying the user selected search type, and determines, based on the search type indicator included in the search request, the selected search type. If the search type indicator indicates that the user selected to include all types of search results (tags, segments, videos and video collections in this example) (i.e., the user has selected to perform a full search), the process 900 proceeds to step 906.

[071] In step 906, an indication or representation of the user's query is determined, such as a query vector. For instance, the terms "Lego" and "Robot" can each be given a term weight using a weighting technique, such as Tf-idf. In this example, the term "Lego" gets the weight 1
and the term "Robot" gets the weight 1.5. Accordingly, the query vector, VQ, for the user's query is:

\[ VQ = (l, 1.5) \]

[072] In step 908, SES 120 selects a set of tags from tag set 192 and determines a vector for each of the selected tags. The selection of the tags may be done based on identifying tags that include one or more of the search terms, or related terms. For example, in step 908, SES 120 selects from tag set 192 every tag included therein that includes at least one of the query terms included in the user's query. In some embodiments, information regarding the tags may be stored by associating each tag, or tag ID, with keywords. In some embodiments, a vector for every tag in tag set 192 may be generated.

[073] In some embodiments, the search terms "Lego" and "Robot" are given a weight for each tag in the database. This may be done, for example, using the same method that was used to give the weights to the query. If the majority of videos in the database being searched have nothing to do with Legos or robots, and do not contain any tags, segments, or other information relating to Legos or robots, the majority of the tags for both terms will have a weight of 0. The vectors for these tags are (0, 0), and all these tags can safely be excluded from the search.

[074] In the non-limiting example process 900, there are three tags in tag set 192 that have non-zero weights for at least one of the query terms. These tags are T1, T2 and T3. These tags may correspond, for instance, to tags T1, T2 and T3 of the example of FIG. 3. Below are their respective vectors, VT1, VT2, and VT3:

\[ VT1 = (1, 0) \]
\[ VT2 = (1, 1) \]
\[ VT3 = (0.5, 1) \]

As shown above, in the tag T1, the terms "Lego" and "robot" have the weights 1 and 0 respectively. Similarly, in tag T2, the terms have the weights 1 and 1, while in tag T3, the terms have the weights 0.5 and 1, respectively.

[075] In step 910, each tag's vector is multiplied by its feature factor. A tag's feature factor depends on whether the tag is present in audio, video, on-screen text etc. In the present
example, it may be assumed that the tags T1, T2 and T3 have the feature factors 0.5, 1 and 2 respectively:

\[ X_1 = 0.5 \]
\[ X_2 = 1 \]
\[ X_3 = 2 \]

For example, T1 may be an image tag, T2 may be a text tag, while T3 is an audio tag. The size of the feature factor may indicate the relative importance of the feature. For instance, in the present example, audio could be twice as important as text, which is twice as important as image. Each tag’s vector is multiplied by its feature factor to yield:

\[ V_{T1}X_1 = (1, 0) \times 0.5 = (0.5, 0) \]
\[ V_{T2}X_2 = (1, 1) \times 1 = (1, 1) \]
\[ V_{T3}X_3 = (0.5, 1) \times 2 = (1, 2) \]

[076] In step 912, an aggregate vector for each segment, video, and video collection (VC) in the corpus is calculated by adding all relevant vectors. For example, it can be assumed that T1, T2 and T3 are all linked to the same video, V, and that T1 and T2 are linked to segment S1, whereas T3 is linked to segment S2. In some embodiments, the vector for a video or segment can be the sum of the tag vectors for the tags that are linked to the video or segment (after multiplication with the tag’s feature factor). So, for example, the aggregate video vector \( W \) for the video V can be:

\[ W = V_{T1}X_1 + V_{T2}X_2 + V_{T3}X_3 = (0.5, 0) + (1, 1) + (1, 2) = (2.5, 3) \]

Similarly, the aggregate segment vectors VS1 and VS2 for segments S1 and S2, respectively, may be the sum of the vectors for all tags present in the segments:

\[ VS_1 = V_{T1}X_1 + V_{T2}X_2 = (0.5, 0) + (1, 1) = (1.5, 1) \]
\[ VS_2 = V_{T3}X_3 = (1, 2). \]

Likewise, an aggregate video collection (VC) vector, denoted VVC, is the sum of the video vectors for the videos included in the VC. For instance, if a particular video collection "a", denoted VCa, consists of videos X, Y, and Z, then the aggregate vector for VCa, denoted WCa,
will be: $V^x + V^y + W^z$, where $W^x$, $V^y$, and $V^z$ are the video vectors for videos $X$, $Y$, and $Z$, respectively.

In step 914, the cosine similarity between the query's vector and each tag's, segment's, video's, video collection's vector is calculated to produce a search score for each tag, segment, video and video collection. The procedure for determining the cosine similarity is given above and results in a search score between 0 and 1 for each tag, each segment and each video. Each tag, segment and video can be considered a separate search result that may be returned to a user. For example, the cosine similarity between the video's vector and the query's vector, i.e. the final search score for the video, is calculated below:

$$RV = (VQ - W) / (||VQ|| ||W||)$$

$$= ((1, 1.5) - (2.5, 3)) / ||(1, 1.5)|| 11(2.5, 3)||$$

$$= (1 * 2.5 + 1.5 * 3) / (V(12 + 1.52) * V(2.52 + 32))$$

$$= 7 / (V(3.25) * V(15.25))$$

$$\sim 0.994.$$

In step 916, the tags, segments, videos and video collections included in the search are ordered, for instance, by descending search score.

In step 918, the list produced in step 916 is presented to the user as the final search result. The list may be transmitted, for example, as a markup language document, such as an HTML or XML document including hypertext transfer protocol links to the content represented in the list.

According to some embodiments, as described above with reference to FIG. 12, the user can also decide if the search result is a list of tags, segments, video, video collections or any combination thereof. For example, a user may choose to search only for videos and segments. If the user selects to only include segments and videos in the search results provided by the process 900, the process proceeds to step 956 where the query's vector is calculated. The process is similar to that described with respect to steps 906-918.

In step 958, the vector of each tag in the corpus is calculated based on the search terms in the query. In step 960, each tag's vector is multiplied by its feature factor. In step 962, an
aggregate vector for each segment and video in the corpus is calculated by adding all relevant tags’ vectors. In step 964, the cosine similarity between the query’s vector and each segment’s and video’s vector is calculated. This is the final search score. In this example, because of the search type, tags are not included in this step. Similarly, no cosine similarity is calculated between the query and the individual tags, since the user has chosen not to list tags in the search result. In this use case, only the segments and videos will be considered separate search results. In step 966, the segments and videos included in the search are ordered by descending search score. In step 968, the list produced in step 966 is presented to the user as the final search result.

[082] FIG. 10 Illustrates an embodiment of SES 120. As shown in FIG. 10, SES 120 may include: a data processing system 1002, which may include one or more general purpose microprocessors and/or one or more circuits, such as an application specific integrated circuit (ASIC), field-programmable gate arrays (FPGAs), and the like; a network interface 1004 configured to enable communication with one or more remote devices via network 110, and a data storage system 1006, which may include one or more non-volatile storage devices and/or one or more volatile storage devices (e.g., random access memory (RAM)). As illustrated, tag set 192 may be stored in data storage system 1006.

[083] In embodiments where data processing system 1002 includes a microprocessor, a computer program product (CPP) 1050 may be provided. CPP 1050 includes a computer readable medium 1051 storing a computer program (CP) 1030 with computer readable instructions/program code. CRM 1051 may be a non-transitory computer readable medium, such as, but not limited, to magnetic media (e.g., a hard disk), optical media (e.g., a DVD), memory devices (e.g., random access memory), and the like, where the non-transitory CRM 1051 is a part of the data storage system 1006. In some embodiments, CP 1030 is configured such that when executed by data processing system 1002, the code causes the data processing system 1002 to perform steps described above (e.g., steps described above with reference to the flow chart shown in FIGs. 5-7 and 9). In other embodiments, SES 120 may be configured to perform steps described herein without the need for code. That is, for example, data processing system 1002 may consist merely of one or more ASICs. Hence, the features of the embodiments described herein may be implemented in hardware and/or software. For example, in particular embodiments, the functional components of the search described above may be implemented by
data processing system 1010 executing computer instructions, by data processing system 1010 operating independent of any computer instructions, or by any suitable combination of hardware and/or software.

According to some embodiments, the processes described herein may be performed by a search engine apparatus 1100 (see FIG. 11). As illustrated in FIG. 11, the search engine apparatus may include, for instance, a number of hardware units, each adapted to perform one or more of the above steps. For example, search engine apparatus 1100 may include a receiver unit 1102 configured to receive, from a client device 101, a search request that includes one or more query terms. The apparatus 1100 may also include a vector determining unit 1104 configured to determine a query vector as well as tag, segment, and/or video vectors as described above in connection with processes 500, 600, 700, and 900. A search score calculating unit 1106 may be configured to calculate search scores based on comparisons between the tag, segment, and/or video vectors and the query vector, as described herein. For example, tag vectors may be generated for a subset of the tags included in tag set 192 and, for each said tag vector, search score calculating unit 1106 calculates a search score for the tag using as inputs the query vector and the tag vector. A search result generating unit 1108 may use the search scores produced by search score calculating unit 1106 to generate a search result. For instance, as discussed above, a hyperlink corresponding to a particular item (e.g., a particular tag, segment, video, etc.) may be included in the search result if the search score for the item exceeds a threshold. The generated search result (e.g., markup language document) may be provided to a transmitter unit 1110 that is operable to transmit the search result towards the client that submitted the search request.

While various embodiments of the present invention have been described above, it should be understood that they have been presented by way of example only, and not limitation. Thus, the breadth and scope of the present invention should not be limited by any of the above-described exemplary embodiments. Moreover, any combination of the above-described elements in all possible variations thereof is encompassed by the invention unless otherwise indicated herein or otherwise clearly contradicted by context.

Additionally, while the processes described above and illustrated in the drawings are shown as a sequence of steps, this was done solely for the sake of illustration. Accordingly, it
is contemplated that some steps may be added, some steps may be omitted, the order of the steps may be re-arranged, and some steps may be performed in parallel.
Claims:

1. A method performed by a search engine system (120), SES, the method comprising:
   receiving (510), at the SES, a search request transmitted by a client device (110a-c),
   wherein said search request includes one or more query terms;
   determining (520), by the SES, a query vector based on said one or more query terms;
   determining (530), by the SES, a first segment vector for a first segment of a first non-
   textual content item;
   determining (540), by the SES, a second segment vector for a second segment of said
   first non-textual content item;
   calculating (550), by the SES, a first segment search score based on the result of a
   comparison of said first segment vector to said query vector; and
   calculating (560), by the SES, a second segment search score based on a result of a
   comparison of said second segment vector to said query vector.

2. The method of claim 1, wherein
   said first segment is associated with a first set of tags,
   said second segment is associated with a second set of tags,
   the first set of tags includes the first tag,
   the second set of tags includes the second tag,
   the method further comprises: determining a first set of tag vectors based on said first set
   of tags and said one or more query terms; and determining a second set of tag vectors based on
   said second set of tags and said one or more query terms,
   the step of determining said first segment vector comprises summing said first set of tag
   vectors, and
   the step of determining said second segment vector comprises summing said second set
   of tag vectors.

3. The method of any of claims 1-2, further comprising:
   determining, by the SES, a first item vector for said first non-textual content item;
   determining, by the SES, a second item vector for a second non-textual content item;
determining, by the SES, a first item search score, wherein said first item search score is based on a comparison of said first item vector to said query vector;

determining, by the SES, a second item search score, wherein said second item search score is based on a comparison of said second item vector to said query vector; and

selecting one or more of: said first segment, said second segment, said first non-textual content item, and said second non-textual content item based on said first segment search score, second segment search score, first item search score, and second item search score.

4. The method of any of claims 1-3, further comprising:

transmitting (580) an ordered set of two or more search results based on the search request, wherein said ordered set of search results includes a first search result that comprises information identifying said first segment, wherein the position of said first search result within said ordered set of search results is determined based on said first segment search score and a search score associated with each other search result included in said ordered set of search results.

5. A method performed by a search engine system (120), SES, the method comprising:

receiving (510), at the SES, a search request transmitted by a client device (110a-c), wherein said search request includes one or more query terms;

determining (520), by the SES, a query vector based on said one or more query terms;

determining (530), by the SES, a first tag vector based on said one or more query terms and a first tag, wherein said first tag is linked with a first feature located in a first segment of anon-textual content item;

determining (540), by the SES, a second tag vector based on said one or more query terms and a second tag, wherein said second tag is linked with a second feature located in a second segment of the non-textual content item;

calculating (550), by the SES, a first tag search score based on the result of a comparison of said first tag vector to said query vector; and

calculating (560), by the SES, a second tag search score based on the result of a comparison of said second tag vector to said query vector.
6. The method of claim 5, wherein
said first set of tag vectors comprises a first weighted tag vector,
determining said first set of tag vectors comprises determining a first initial tag vector for the first tag and multiplying said first initial tag vector with a featurescore associated with a feature type of the first tag, thereby producing the first weighted tag vector, and
the feature type is one of image, audio, video, and text.

7. The method of any of claims 5-6, wherein said search request includes a search type indicator that indicates a user is requesting, at the least, a tag search.

8. A search engine system (120), SES, comprising:
adata storage system (1006) and a data processing system (1002), said data storage system comprising instructions executable by the data processing system whereby the SES is operative to:
determine a query vector based on query terms included in a search request;
determine a first segment vector for a first segment of a first non-textual content item;
determine a second segment vector for a second segment of said first non-textual content item;
calculate a first segment search score based on the result of a comparison of said first segment vector to said query vector; and
calculate a second segment search score based on a result of a comparison of said second segment vector to said query vector.

9. The SES of claim 8, wherein the SES is operative to:
calculatesaid first segment search score by, at least, calculating: \( \frac{V \cdot VS1}{||V|| \cdot ||VS1||} \), where VQ is said query vector, and VS1 is said first segment vector, and
calculatesaid second segment search score by, at least, calculating: \( \frac{V \cdot VS2}{||V|| \cdot ||VS2||} \), where VS2 is said second segment vector.

10. The SES of claim 8 or 9, wherein
said first segment is associated with a first set of tags and said second segment is associated with a second set of tags, wherein the first set of tags includes a first tag and the second set of tags includes a second tag, and

the SES is operative to:

determine a first set of tag vectors based on said first set of tags and said one or more query terms,

determine a second set of tag vectors based on said second set of tags and said one or more query terms,

determinesaid first segment vector by summing said first set of tag vectors, and
determine said second segment vector by summing said second set of tag vectors.

11. The SES of any of claims 8-10, wherein the SES is operative to:

determine a first item vector for said first non-textual content item;

determine a second item vector for a second non-textual content item;

determine a first item search score, wherein said first item search score is based on a comparison of said first item vector to said query vector; and

determine a second item search score, wherein said second item search score is based on a comparison of said second item vector to said query vector.

12. The SES of claim 11, wherein the SES is operative to:

select one or more of: said first segment, said second segment, said first non-textual content item, and said second non-textual content item based on said first segment search score, second segment search score, first item search score, and second item search score.

13. The SES of any of claims 10-12, wherein

said first set of tag vectors comprises a first weighted tag vector,

the SES is operative to determine said first set of tag vectors by determining a first initial tag vector for the first tag and multiplying said first initial tag vector with a features value associated with a feature type of the first tag, thereby producing the first weighted tag vector, and

the feature type is one of image, audio, video, and text.
14. The SES of any of claims 8-13, wherein the SES is operative to:
transmit an ordered set of two or more search results based on the search request, wherein said ordered set of search results includes a first search result that comprises information identifying said first segment, wherein the position of said first search result within said ordered set of search results is determined based on said first segment search score and a search score associated with each other search result included in said ordered set of search results.

15. The SES of any of claims 8-14, wherein said search request includes a search type indicator that indicates a user is requesting a segment search.

16. A search engine system (120), SES, comprising:
   a data storage system (1006) and a data processing system (1002), said data storage system comprising instructions executable by the data processing system whereby the SES is operative to:
   determine a query vector based on query terms included in a search request;
   determine a first tag vector based on said one or more query terms and a first tag, wherein said first tag is linked with a first feature located in a first segment of a non-textual content item;
   determine a second tag vector based on said one or more query terms and a second tag, wherein said second tag is linked with a second feature located in a second segment of the non-textual content item;
   calculate a first tag search score based on the result of a comparison of said first tag vector to said query vector; and
   calculate a second tag search score based on the result of a comparison of said second tag vector to said query vector.

17. The SES of claim 16, wherein the SES is operative to:
calculate said first tag search score by, at least, calculating: \((VQ \cdot VT1) / (||VQ|| \cdot ||VT1||)\), where VQ is said query vector, and VT1 is said first tag vector, and
determine said second tag search score by, at least, calculating: \((VQ \cdot VT2) / (||VQ|| \cdot ||VT2||)\), where VT2 is said second tag vector.
18. The SES of any of claims 16-17, wherein the SES is operative to:
   determine a first item vector for said first non-textual content item;
   determine a second item vector for a second non-textual content item;
   determine a first item search score, wherein said first item search score is based on a
   comparison of said first item vector to said query vector; and
   determine a second item search score, wherein said second item search score is based on
   a comparison of said second item vector to said query vector.

19. The SES of claim 18, wherein the SES is operative to:
   select one or more of: said first tag, said second tag, said first non-textual content item,
   and said second non-textual content item based on said first tag search score, second tag search
   score, first item search score, and second item search score.

20. The SES of any one of claims 16-16, wherein the SES is operative to:
   determine the first tag vector by determining an first initial tag vector and multiplying
   said first initial tag vector with a featurescore associated with a feature type of the first tag.

21. The SES of any of claims 16-20, wherein the SES is operative to:
   transmit an ordered set of two or more search results based on the search request, wherein
   said ordered set of search results includes a first search result that comprises information
   identifying said first tag, wherein the position of said first search result within said ordered set of
   search results is determined based on said first tag search score and a search score associated
   with each other search result included in said ordered set of search results.

22. The SES of any of claims 16-21, wherein said search request includes a search type
   indicator that indicates a user is requesting a tag search.

23. A search engine apparatus (1100), comprising:
   a receiver unit (1102) configured to receive a search request transmitted by a client
device (110), wherein said search request includes one or more query terms;
   a vector determining unit (1104) configured to:
determine a query vector based on said one or more query terms;

determine a first segment vector for a first segment of a first non-textual content item; and

determine a second segment vector for a second segment of said first non-textual content item; and

a search score calculating unit (1106) configured to:
calculate a first segment search score based on the result of a comparison of said first segment vector to said query vector; and

calculate a second segment search score based on a result of a comparison of said second segment vector to said query vector.

24. A search engine apparatus (1100), comprising:
a receiver unit (1102) configured to receive a search request transmitted by a client device (110), wherein said search request includes one or more query terms;
vector determining unit (1104) configured to:
determine a query vector based on said one or more query terms;
determine a first tag vector based on said one or more query terms and a first tag, wherein said first tag is linked with a first feature located in the first segment of the non-textual content item; and
determine a second tag vector based on said one or more query terms and a second tag, wherein said second tag is linked with a second feature located in the second segment of the non-textual content item; and

a search score calculating unit (1106) configured to:
calculate a first tag search score based on the result of a comparison of said first tag vector to said query vector; and

calculate a second tag search score based on the result of a comparison of said second tag vector to said query vector.

25. A computer program product (1050) comprising a non-transitory computer readable medium (1051) storing computer instructions (1030) for searching content, the computer instructions (1030) comprising:
instructions for determining a query vector based on query terms included in a search request;
instructions for determining a first segment vector for a first segment of a first non-textual content item;
instructions for determining a second segment vector for a second segment of said first non-textual content item;
instructions for calculating a first segment search score based on the result of a comparison of said first segment vector to said query vector; and
instructions for calculating a second segment search score based on the result of a comparison of said second segment vector to said query vector.

26. A computer program product (1050) comprising a non-transitory computer readable medium (1051) storing computer instructions (1030) for searching content, the computer instructions (1030) comprising:
instructions for determining a query vector based on query terms included in a search request;
instructions for determining a first tag vector based on said one or more query terms and a first tag, wherein said first tag is linked with a first feature located in the first segment of the non-textual content item;
instructions for determining a second tag vector based on said one or more query terms and a second tag, wherein said second tag is linked with a second feature located in the second segment of the non-textual content item;
instructions for calculating a first tag search score based on the result of a comparison of said first tag vector to said query vector; and
instructions for calculating a second tag search score based on the result of a comparison of said second tag vector to said query vector.

27. A computer program (1030) comprising computer readable instructions which when run on a search engine system (120) causes the search engine system (120) to:
determine a query vector based on query terms included in a received search request;
determine a first segment vector for a first segment of a first non-textual content item;
determine a second segment vector for a second segment of said first non-textual content item;
calculate a first segment search score based on the result of a comparison of said first segment vector to said query vector; and
calculate a second segment search score based on a result of a comparison of said second segment vector to said query vector.

28. A computer program(1030) comprising computer readable instructions which when run on a search engine system (120) causes the search engine system (120) to:
      determine a query vector based on query terms included in a received search request;
      determine a first tag vector based on said one or more query terms and a first tag, wherein said first tag is linked with a first feature located in the first segment of the non-textual content item;
      determine a second tag vector based on said one or more query terms and a second tag, wherein said second tag is linked with a second feature located in the second segment of the non-textual content item;
      calculate a first tag search score based on the result of a comparison of said first tag vector to said query vector; and
      calculate a second tag search score based on the result of a comparison of said second tag vector to said query vector.
500

510 Receive a Search Request that Includes One or More Query Terms

520 Determine a Query Vector Based on the Query Terms

530 Determine a First Tag Vector Based on the Query Terms and a First Tag

540 Determine a Second Tag Vector Based on the Query Terms and a Second Tag

550 Determine a First Tag Search Score Based on a Comparison of the Query Vector and the First Tag Vector

560 Determine a Second Tag Search Score Based on a Comparison of the Query Vector and the Second Tag Vector

570 Generate a search result for the search request

580 Transmit the search results to the user

FIG. 5
600

620 Receive a Search Request that Includes One or More Query Terms

620 Determine a Query Vector Based on the Query Terms

630 Determine a First Segment Vector for a First Segment

640 Determine a Second Segment Vector for a Second Segment

650 Determine a First Segment Search Score Based on a Comparison of the Query Vector and the First Segment Vector

660 Determine a Second Segment Search Score Based on a Comparison of the Query Vector and the Second Segment Vector

670 Generate a search result for the search request

680 Transmit Search Results to the User

FIG. 6
Receive a Search Request that Includes One or More Query Terms

Determine a Query Vector Based on the Query Terms

Determine a First Video Vector for a First Video

Determine a Second Video Vector for a Second Video

Determine a First Video Search Score Based on a Comparison of the Query Vector and the First Video Vector

Determine a Second Video Search Score Based on a Comparison of the Query Vector and the Video Segment Vector

Generate a search result for the search request

Transmit the search Result to the User

FIG. 7
<table>
<thead>
<tr>
<th><strong>Video</strong></th>
<th><strong>Segment</strong></th>
<th><strong>TAG</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_1$</td>
<td>$S_1$</td>
<td>$T_1, T_2 \ldots T_N$</td>
</tr>
<tr>
<td></td>
<td>$S_2$</td>
<td>$T_1, T_2 \ldots T_N$</td>
</tr>
<tr>
<td></td>
<td>$S_3$</td>
<td>$T_1, T_2 \ldots T_N$</td>
</tr>
<tr>
<td></td>
<td>$S_4$</td>
<td>$T_1, T_2 \ldots T_N$</td>
</tr>
<tr>
<td>$V_2$</td>
<td>$S_1$</td>
<td>$T_1, T_2 \ldots T_N$</td>
</tr>
<tr>
<td></td>
<td>$S_2$</td>
<td>$T_1, T_2 \ldots T_N$</td>
</tr>
<tr>
<td>$V_N$</td>
<td>$S_N$</td>
<td>$T_1, T_2 \ldots T_N$</td>
</tr>
</tbody>
</table>

**FIG. 8**
User Enters Query Terms, selects search type, and submits the query

Receive search request

Include All Types

Determine Query Vector

Determine Vector of Each Tag in Corpus

Multiply Tag Vectors by Feature Factors

Determine Aggregate Vector for Each Segment, Video and Video Collection (VC)

Determine Cosine Similarity Between Query Vector and Each Tag, Segment, Video, VC Vector

Tags, Segments, Videos, VCs Ordered By Search Score

Result Transmitted to the User

Limited to Segment and Video

Determine Query Vector

Determine Vector of Each Tag in Corpus

Multiply Tag Vectors by Feature Factors

Determine Aggregate Vector for Each Segment and Video

Determine Cosine Similarity Between Query Vector and Each Segment and Video Vector

Segments and Videos Ordered By Search Score

Result Transmitted to the User

FIG. 9
FIG. 12
### A. CLASSIFICATION OF SUBJECT MATTER

**IPC:** see extra sheet

According to International Patent Classification (IPC) or to both national classification and IPC

### B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

**IPC:** G06F

Documentary searched other than minimum documentation to the extent that such documents are included in the fields searched

SE, DK, FI, NO classes as above

Electronic database consulted during the international search (name of database and, where practicable, search terms used)

EPO-Internal, PAJ, WPI data, COMPENDEX, INSPEC

### C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category*</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>EP 1024437 A2 (XEROX CORP), 2 August 2000 (2000-08-02); abstract; page 1, paragraph [0001] - page 5, paragraph [0033]; page 6, paragraph [0041] - page 13, paragraph [0079]; page 15, paragraph [01 01] - page 16, paragraph [01 02]</td>
<td>1-28</td>
</tr>
</tbody>
</table>


* Further documents are listed in the continuation of Box C.  
 See patent family annex.

* Special categories of cited documents:
  * "A" document defining the general state of the art which is not considered to be of particular relevance  
  * "E" earlier application or patent but published on or after the international filing date  
  * "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)  
  * "O" document referring to an oral disclosure, use, exhibition or other means  
  * "P" document published prior to the international filing date but later than the priority date claimed  
  * "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention  
  * "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone  
  * "Y" document of particular relevance; the claimed invention cannot be considered inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art  
  * "&" document member of the same patent family

**Date of the actual completion of the international search:** 09-04-2014  
**Date of mailing of the international search report:** 10-04-2014

**Name and mailing address of the ISA/SE**  
Patent- och registreringssverket  
Box 5055  
S-102 42 STOCKHOLM  
Facsimile No. +46 8 666 02 86

**Authorized officer**  
Jimmie Femzen  
Telephone No. +46 8 782 25 00

Form PCT/ISA/210 (second sheet) (July 2009)
<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>&quot;Discriminative tag learning on youtube videos with latent sub-tags&quot; Yang et al</td>
<td>1-28</td>
</tr>
</tbody>
</table>


---

Form PCT/ISA/210 (continuation of second sheet) (July 2009)
Continuation of: second sheet

International Patent Classification (IPC)

G06F 17/30 (2006.01)
<table>
<thead>
<tr>
<th>Country</th>
<th>Publication Number</th>
<th>Date</th>
<th>Application Number</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>EP</td>
<td>1024437 A2</td>
<td>02/08/2000</td>
<td>DE 60044220</td>
<td>02/06/2010</td>
</tr>
<tr>
<td>EP</td>
<td>2178007 A3</td>
<td>25/08/2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EP</td>
<td>2178006 A3</td>
<td>13/04/2011</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EP</td>
<td>2284733 A3</td>
<td>25/05/2011</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EP</td>
<td>2178008 A3</td>
<td>01/09/2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JP</td>
<td>4854799 B2</td>
<td>18/01/2012</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JP</td>
<td>2011253570 A</td>
<td>15/12/2011</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JP</td>
<td>2010267277 A</td>
<td>25/11/2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JP</td>
<td>2010262662 A</td>
<td>18/11/2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JP</td>
<td>2010250849 A</td>
<td>04/11/2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JP</td>
<td>2010218579 A</td>
<td>30/09/2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JP</td>
<td>2010205306 A</td>
<td>16/09/2010</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JP</td>
<td>2000339350 A</td>
<td>08/11/2000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>JP</td>
<td>4874413 B2</td>
<td>15/02/2011</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Form PCT/ISA/210 (patent family annex) (July 2009)