TABLE OF CONTENTS FOR SEARCH QUERY REFINEMENT

BACKGROUND

[0001] The current state of user search interfaces relies primarily on query parsing and matching to large lists of potential results. The user is burdened with sifting through these large lists to obtain the information he or she is seeking. The search results presented to the user are often muddled by long complex user queries, which increase the time required to generate the search results and often result in a strange amalgam of results related to individual terms within the complex query.

SUMMARY

[0002] Embodiments of the present invention generally relate to systems, methods, and computer-storage media having computer-executable instructions embodied thereon that, when executed, perform methods for providing a user interface for exploring facets relevant to a user search query term, wherein a state of the user search query is maintained. Utilizing the systems, methods, and media described herein, a user search query is received at a front end engine, which sends the user search query to a back end server. The back end server generates facets, or categories of relevant topics, which are received by the front end engine to be rendered into a Table of Contents or TOC. The TOC displays these facets to be selected by the user and once a facet is selected, the selected facet is sent to the back end server. Together with the recalled original search query term, the selected facet is used to generate a refined search results list. The user may continue to browse the refined search result list or select another facet of the original search query term to explore. Since the state of the original search query is maintained throughout the browsing session, a user may explore various aspects of the search query term using the facets presented in the TOC. In alternate embodiments, the TOC may include a set of sub facets or sub categories of each facet to further refine the search query term. In other embodiments, the TOC may incorporate static facets to provide general categories for refining the browsing session.

[0003] Embodiments of the invention are defined by the claims below, not this summary. A high-level overview of various aspects of the invention are provided here for that reason, to provide an overview of the disclosure, and to introduce a selection of concepts that are further described below in the detailed-description section below. This summary is not intended to identify key features or essential features of the claimed subject matter, nor is it intended to be used as an aid in isolation to determine the scope of the claimed subject matter.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

[0004] Illustrative embodiments of the present invention are described in detail below with reference to the attached drawing figures, and wherein:

[0005] FIG. 1 is a block diagram of an exemplary computing environment suitable for use in implementing embodiments of the present invention.

[0006] FIG. 2 is a flow diagram illustrating an embodiment of the method of providing a user interface for exploring facets relevant to a user search query term, wherein a state of the user search query is maintained.

[0007] FIG. 3 is a flow diagram illustrating an alternative embodiment of the method of providing a user interface for exploring facets relevant to a user search query term, wherein a state of the user search query is maintained.

[0008] FIG. 4A-B are block diagrams illustrating the functional components of a computerized system for providing a user interface for exploring facets relevant to a user search query, wherein a state of the user search query is maintained.

[0009] FIG. 5A-B are block diagrams illustrating an alternative embodiment of a computerized system for providing a user interface for exploring facets relevant to a user search query, wherein a state of the user search query is maintained.

[0010] FIG. 6A-D are an exemplary operating environment for the present invention which provides a user interface for exploring facets related to a search query term while maintaining the state of the original search query term.

[0011] FIG. 7A-D are an alternative embodiment of the operating environment for the present invention, which includes the use of sub facets to further refine the user search experience.

[0012] FIG. 8A-B are an additional embodiment of the operating environment for the present invention including the incorporation of static facets.

DETAILED DESCRIPTION

[0013] The subject matter of embodiments of the present invention is described with specificity herein to meet statutory requirements. But the description itself is not intended to necessarily limit the scope of claims. Rather, the claimed subject matter might be embodied in other ways to include different steps or combinations of steps similar to the ones described in this document, in conjunction with other present or future technologies. Terms should not be interpreted as implying any particular order among or between various steps herein disclosed unless and except when the order of individual steps is explicitly described.

[0014] The present invention relates generally to providing a user interface for exploration and refinement of user generated queries. Users are moving away from employing single-query searches and pursuing more extended query sessions. Attempting to explore aspects of a query using a standard search leads to reduced user satisfaction with search result accuracy. Users often resort to long complex queries to force more relevant search results; however, this strategy generally does not produce the desired effect. The Table of Contents or TOC provides a user interface wherein the user inputs a search query term and is provided with search query term specific facets, which categorize and organize various aspects of the search query term. The user may then explore and navigate these facets while staying within the original topic of the search query term. Selection of the facets refines the search query term and may be used to generate more focused and relevant search results.

[0015] In a first aspect of the invention, a set of computer-useable instructions provide a method for providing a TOC Navigation Model or user interface to be executed on a processor of a computing device. The method in one embodiment includes receiving at least one search query term generated by a user. The search query term is used to generate at least one facet, categorizing topics relevant to the search query term. The generated facets are received and displayed to the user. The user may select a particular facet he or she wishes to explore further and the selected facet with the original search query term is used to generate a refined search result list, which is displayed to the user. The user may then choose to explore these search results or continue selecting and navigating through the facets of the original search query term. Since the original search query term or the state is maintained, the user may easily explore all facets of the search query term.

[0016] In another aspect of the invention, a set of computer-useable instructions provide a method for providing a TOC user interface to be executed on a processor of a computing device. The method in one embodiment includes receiving at least one search query term generated by a user. The search query term is used to generate at least one facet, categorizing topics relevant to the search query tem. The generated facets are received and displayed to the user. The user may select a particular facet he or she wishes to explore further and the selected facet with the original search query term is used to generate at least one sub facet, which is displayed to the user. The user may choose to select a sub facet, which is used with the selected facet and the search query term to generate a refined search results list. The user may then choose to explore the displayed search results for the selected sub facet or select another sub facet. In one embodiment,

the user may select a new facet to display a new set of relevant sub facets. Since the original search query term or the state is maintained, the user may easily explore all facets and all sub facets of the search query term.

[0017] In another embodiment of the present invention, a computerized system provides the facet generation and maintenance of the search state for the TOC navigation model. A front end engine receives the search query term and passes this to a back end server. The back end server may in one embodiment store the search query term. In another embodiment, the front end engine may re-send the search query term with each navigation step. Once the search query term is received by the back end server, a number of algorithms are used to generate facets, which categorize topics relevant to the search query term. These generated facets are returned to the front end engine, which displays these facets in the user interface in a TOC. The user may select a facet to explore further which is sent to the back end server. In one embodiment, the back end server may recall the original search query term from a local database. In another embodiment, the front end engine may re-send the original search query term with the selected facet to the back end engine. In either case, the state of the original user query is maintained. Using the selected facet and the search query term, the back end server generates a refined list of search results. These search results are returned to the front end engine, which renders the search result in the TOC for the user to peruse. The user may choose to explore these search results or continue selecting other facets to navigate.

[0018] In another embodiment of the present invention, a computerized system provides the facet and sub facet display and generation as well as the maintenance of the search state for the TOC navigation model. A front end engine receives the search query term and passes this to a back end server. The back end server may in one embodiment store the search query term. In another embodiment, the front end engine may re-send the search query term with each navigation step. Once the search query term is received by the back end server, a number of algorithms are used to generate facets, which categorize topics relevant to the search query term. These generated facets are returned to the front end engine, which displays these facets in the user interface in a TOC. The user may select a facet to explore further which is sent to the back end server. In one embodiment, the back end server may recall the original search query term from a local database. In another embodiment, the front end engine may re-send the original search query term with the selected facet to the back end engine. In either case, the state of the original user query is maintained. Using the selected facet and the search query term, the back end server

generates a set of sub facets. These sub facets are returned to the front end engine, which renders the sub facets in the TOC for the user to peruse. The user may choose to explore these sub facets or continue selecting other facets to navigate. If the user selects a sub facet, the front end engine sends the selected sub facet to the back end server. The selected sub facet, the selected facet, and the search query term are used to generate a refined search result list, which is returned to the front end engine for TOC display. The user may choose to explore the search result list for the selected sub facet or continue navigating through facets for the original search query term.

[0019] In one aspect of the invention, the TOC model may be displayed in a user interface embodied on a web browser. The search results provided to the user may be a set of web links for navigating internet sites relevant to the search query term. In another aspect of the invention, the TOC navigation model may include a set of static facets, which may also be employed to refine the search query term.

[0020] Having briefly described in an overview of the present invention, an exemplary operating environment in which various aspects of the present invention may be implemented is now described. Referring to the drawings in general, and initially to FIG. 1 in particular, an exemplary operating environment for implementing embodiments of the present invention is shown and designated generally as computing device 100. Computing device 100 is but one example of a suitable computing environment and is not intended to suggest any limitation as to the scope of use or functionality of the invention. Neither should the computing device 100 be interpreted as having any dependency or requirement relating to any one or combination of components illustrated.

[0021] Embodiments of the invention may be described in the general context of computer code or machine-useable instructions, including computer-executable instructions such as program modules, being executed by a computer or other machine, such as a personal data assistant or other handheld device. Generally, program modules including routines, programs, objects, components, data structures, etc., refer to code that perform particular tasks or implement particular abstract data types. Embodiments of the invention may be practiced in a variety of system configurations, including hand-held devices, consumer electronics, general-purpose computers, more specialty computing devices, and the like. Embodiments of the invention may also be practiced in distributed computing environments where tasks are performed by remote-processing devices that are linked through a communications network.

[0022] With continued reference to FIG. 1, computing device 100 includes a bus 110 that directly or indirectly couples the following devices: memory 112, one or more processors 114, one or more presentation components 116, input/output (I/O) ports 118, I/O components 120, and an illustrative power supply 122. Bus 110 represents what may be one or more busses (such as an address bus, data bus, or combination thereof). Although the various blocks of FIG. 1 are shown with lines for the sake of clarity, in reality, delineating various components is not so clear, and metaphorically, the lines would more accurately be grey and fuzzy. For example, one may consider a presentation component such as a display device to be an I/O component. Additionally, many processors have memory. The inventors hereof recognize that such is the nature of the art, and reiterates that the diagram of FIG. 1 is merely illustrative of an exemplary computing device that can be used in connection with one or more embodiments of the present invention. Distinction is not made between such categories as "workstation," "server," "laptop," "hand-held device," etc., as all are contemplated within the scope of FIG. 1 and reference to "computing device."

[0023] Computing device 100 typically includes a variety of computer-readable media. Computer-readable media can be any available media that can be accessed by computing device 100 and includes both volatile and nonvolatile media, removable and non-removable media. By way of example, and not limitation, computer-readable media may comprise computer-storage media and communication media. Computer-storage media includes both volatile and nonvolatile, removable and non-removable media implemented in any method or technology for storage of information such as computer-readable instructions, data structures, program modules or other data. Computer-storage media includes, but is not limited to, Random Access Memory (RAM), Read Only Memory (ROM), Electronically Erasable Programmable Read Only Memory (EEPROM), flash memory or other memory technology, CD-ROM, digital versatile disks (DVD) or other holographic memory, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or any other medium that can be used to encode desired information and which can be accessed by the computing device 100.

[0024] The memory 112 includes computer-storage media in the form of volatile and/or nonvolatile memory. The memory 112 may be removable, non-removable, or a combination thereof. Exemplary hardware devices include solid-state memory, hard drives, optical-disc drives, etc. The computing device 100 includes one or more processors that read data from various entities such as the memory 112 or the I/O

components 120. The presentation component(s) 116 present data indications to a user or other device. Exemplary presentation components include a display device, speaker, printing component, vibrating component, and the like.

[0025] The I/O ports 118 allow the computing device 100 to be logically coupled to other devices including the I/O components 120, some of which may be built in. Illustrative I/O components 120 include a microphone, joystick, game pad, satellite dish, scanner, printer, wireless device, and the like.

[0026] FIG. 2 is a flow chart depicting an illustrative method for providing a user interface for exploring facets relevant to a user search query term, wherein a state of the user search query is maintained. A search query term may be any subject that the user wishes to explore or research. Typically, the search query term may be quite broad and yield thousands of results that are organized into facets. A facet may be any categorization of relevant topics relevant to the user search query term. Facets used to organize the typically enormous and often chaotic search result list generated by search engines. Generally, in a search environment wherein the state is not maintained or where the state is not apparent to the user, the user enters a search query term and a set of search results are returned. As the user selects a desired search term to investigate further, the state of the search navigates away from the original search query term. In this manner, the state is not maintained, or if it is maintained, it is not apparent to the user. In the exemplary method, the user can navigate between various facets while maintaining the state of the original search query term. The computer useable instructions for performing this embodiment 200 may be written on a computer readable media and executed on a processor of a computing device. The invention is broadly applicable and may be employed on a wide variety of computing devices used for executing the computer useable instructions such as personal computers, super computers, computing clusters, game consoles, mp3 players, electronic books, mobile computing devices, cellular phones, PDAs, etc. The method illustrated in 200 is an embodiment for providing a user interface for exploring facets relevant to a user search query term wherein the facets categorize topics related to the at least one search query term. The facets allow the user to refine the at least one search query term.

[0027] In step 201, at least one search query term is received in one embodiment at a front end engine. The front end engine may be embodied at a computing device providing the user interface or may be accessed remotely via a computing device providing the user interface. The front end engine in one embodiment provides the TOC interface for the

user to interact with the generated facets and search results. The at least one search query term may be applied to a back end server, such as a search engine, or similar device which generates at least one facet or a set of facets based upon dictionary-based algorithms. A search engine utilizes algorithms and human input to search web pages, images, and other types of files. An exemplary search engine may incorporate a web crawler which retrieves a vast set of web pages and then indexes them according to content. The facets are search query term specific and may be selected by any method for determining categories relevant to a search term. Examples of facet selection will be discussed in subsequent embodiments.

[0028] The generated facets for the search query term are received 202 and displayed 203 to the user. In an exemplary method, the front end engine may receive and display the generated facets in the form of a Table of Contents (TOC) Navigation Model or user interface. A TOC may be a graphical or text-based user interface that displays and allows selection of the generated facets and search results by the user during the browsing session. The TOC also provides a browsing session wherein the state of the session is maintained and as the user selects different facets, the original search query term is recalled. The user may select a facet that he or she wishes to explore further 204, for example by clicking on the facet displayed in the TOC. In response to the user selection, a set of search results relevant to the selected facet and the search query term is displayed 205. In an exemplary method, the front end engine may display the TOC with the facets relevant to the search query term and the list of search results relevant to the selected facet and the search query term. The search results may include a list of relevant websites. The user may explore the displayed results or select another relevant facet of the search query term in order to display a new set of search terms relevant to the new selected facet and the search query term. This method demonstrates that the search query term state is maintained throughout the user search experience of navigating facets of the search query term.

[0029] The state of the search query term may be maintained by any number of protocols, for example, the original search query term may be saved in a database associated with the facet generating device or back end server. In an alternative embodiment, an IP address or other source identifier may be associated with the search query term during storage. Upon receiving a selected facet or other user selection, the source identifier of the new selection is matched with the identifier of the original search query term and the matched selected facet and search query term are used to generate

subsequent results at the back end server. In another embodiment, the original search query term may be stored in a front end engine which resends the search query term with each selected facet. In another embodiment, the search query term may be stored in a separate auxiliary database to be recalled when necessary by either the front end engine or the back end server. In either embodiment, any selection of a relevant facet maintains state by recalling the original search query term.

[0030] In an alternative embodiment, in addition to receiving relevant facets for the search query term, a set of search results relevant to the entire search query term may be received and displayed. The user has the choice between using the TOC based facets to explore relevant aspects of the search query term or exploring individual search results relevant to the search query term. The alternative embodiment would allow the user to explore all search results while maintaining the state of the original search query term and quickly narrow down the search results by selecting facets.

[0031] The method may be executed on a computer connected to internet and the display may include a web browser. In one embodiment, the search results may include Uniform Resource Identifiers (URIs) and the facets relevant to a user search query term may categorize the URIs relevant to a particular search query term. In one embodiment the user search query term is sent to a search engine which is configured to generate relevant facets based upon the returned search terms and various dictionary sources. In another embodiment, a set of static facets may be displayed to the user along with the generated relevant facets. For example, static facets may include categories of document types or chronological headings. Static facets are universal headings for search queries that are not specific to the user search query term but serve as additional means for limiting the user search and aiding in the user search experience. In another embodiment, a prompt may be displayed to the user for input of the search query term.

[0032] FIG. 3 is a flow chart illustrating an alternative embodiment of the present invention. The method depicted allows a user to explore facets related to a search query term while maintaining the state of the original search query term. In this exemplary method, sub facets are provided to the user in order to further refine the search query term and produce relevant search results to the user. The computer useable instructions for performing this embodiment 300 may be written on a computer readable media and executed on a processor of a computing device. The invention is broadly applicable and may be employed on a wide variety of computing devices that execute computer useable instructions such as personal computers, super computers, computing clusters, game

consoles, mp3 players, electronic books, mobile computing devices, cellular phones, PDAs, etc.

[0033] The method illustrated in 300 is an embodiment for providing a user interface for exploring facets and sub facets relevant to a user search query term wherein the facets categorize topics related to the at least one search query term. In the exemplary method, the at least one search query term is received 301, for example, by the front end engine. The search query term is then used to generate at least one relevant facet which categorizes a set of topics related to the at least one search query term. The at least one relevant facet may be generated by any means for categorizing relevant topics. In one embodiment, a back end server may receive the search query term and produce a set of facets. The facets may be generated by any method for determining categories relevant to a search query term. Examples of facet selection will be described in following paragraphs.

[0034] The at least one relevant facet for the search query term is received 302 and displayed 303 to the user. In one embodiment, the facets are displayed in the form of a TOC. The user selects at least one relevant facet to refine the search query 304 and at least one sub facet relevant to the selected facet and the search query term is received 305. In one embodiment, a back end server may receive the selected facet and produce a set of sub facets based upon dictionary and user data-seeded algorithms. An exemplary sub facet is a sub category of the category of relevant topics described by the facet. Any facet may have a number of sub facets, the presentation of which may be useful to a user in refining the topic he or she is searching. Once the sub facets for the selected facet and search query term are received 305, the sub facets and facets for the search query term are displayed 306. In one embodiment, the front end engine renders the sub facets and facets into a TOC. The user then selects a sub facet he or she wishes to further explore 307 and a set of search results for the selected sub facet, the selected facet, and the search query term are received 308 and displayed 309. In an exemplary environment, the received data is displayed in a TOC.

[0035] The state of the search query term may be maintained in one embodiment by storing the search query term in a database which is accessed during facet and sub facet generation. Once a user selects a facet, that facet may be stored as well for access during sub facet generation. In an alternative embodiment, the state of the search query term may be maintained by resending the original search query term from the front end engine each time a facet or a sub facet is selected. Once a user selects a facet, that facet may be resent

during sub facet generation. In either embodiment, any selection of a relevant facet maintains state by recalling the original search query term.

[0036] In an alternative embodiment, in addition to receiving relevant facets and sub facets for the search query term, a set of search results relevant to the search query term may be received and displayed. In this embodiment, a user may simply navigate between the displayed search results or choose to refine the displayed results by selecting applicable facets. The alternative embodiment would allow the user to explore all search results while maintaining the state of the original search query term and quickly narrow down the search results by selecting facets and sub facets.

[0037] The method may be executed on a computer connected to internet and the display may include a web browser. In one embodiment, the search results may include Uniform Resource Identifiers and the facets relevant to a user search query term may categorize the URIs relevant to a particular search query term. In one embodiment the user search query term is sent to a search engine which is configured to generate relevant facets based upon the returned search terms and various dictionary sources. In another embodiment, a set of static facets may be displayed to the user along with the generated relevant facets. For example, static facets may include categories of document types or chronological headings. Static facets are universal headings for search queries that are not specific to the user search query term but serve as additional means for limiting the user search and aiding in the user search experience. In another embodiment, a prompt may be displayed to the user for input of the search query term.

[0038] FIG. 4A is a block diagram depicting one embodiment of the claimed invention. A user generated search query term is received at 403 at a front end engine 401. The front end engine may be embodied together with or separately from the back end server 402. In either case, processes performed at the front end engine 401 and the back end server 402 may be written onto a computer readable media and executed on a processor of any computing device. The front end engine 401 sends the search query term at 404 to the back end engine 402 wherein the search query term is stored in a database 405. The back end server 402, such as a search engine, determines what facets or categories are most relevant to the search query term. The back end server 402 includes any algorithms necessary to accomplish this task or any other processes available for facet generation. For example, the back end server 402 may include query logs of categories most accessed in relation to a search query term as well as query specialization algorithms. The back end server may utilize stored dictionaries to establish categorization for a search

query term. Algorithms within the back end server may be employed to break down the dictionary into a smaller more easily accessible set. The dictionary algorithms may be used expand upon categories associated with a search query term or determine synonyms for categories. Various random generation algorithms may also be employed.

[0039] Once the facets are generated at the back end server 402, they are received at 406 by the front end engine 401. In one embodiment a Table of Contents (TOC) generator 407 creates a display of the relevant facets for the user. The user may then select a facet to explore further and the selection of the facet is sent at 408 from the front end engine 401 to the back end engine 405. In order to maintain the state of the user search experience, the original search query term may be recalled from the database 405 and the selected facet and the search query term are used to refine a set of search results. The set of search results for the selected facet and the search query term are sent at 409 and received by the front end engine 401, which renders the TOC 407. The user may explore the displayed search results or select another facet to explore.

[0040] FIG. 4B is a block diagram 410 depicting an alternative embodiment of the present invention. At 413, a user generated search query term is received at a front end engine 411. The front end engine may be embodied together with or separately from the back end service 412. In either case, processes performed at the front end engine 411 and the back end server 412 may be written onto a computer readable media and executed on a processor of any computing device. The search query term is passed at 414 to the back end server 412, which generates at least one facet relevant to the search query term using, in one embodiment, random walk and dictionary based algorithms. The generated facets are returned at 415 to the front end engine 411 and rendered into a TOC 419. The user may select a facet he or she wishes to explore further and the selected facet is returned with the original search query term 416 to the back end server 412. Additional embodiments may include a database for search query term storage at the front end engine 411. The back end server 412 then utilizes the selected facet to refine the search query term to generate a list of search results. The search results of the selected facet and the search query term are returned 417 to the front end engine 411 and displayed in the TOC. The user may then explore these search results or select another facet to explore. In either case the state of the original search query term is maintained and the user does not have to re-enter the original search query term in order to retain the original search parameters.

[0041] FIG. 5A is a block diagram 500 illustrating an alternative embodiment of the present invention. This embodiment also includes a front end engine 501 and a back end server 502. The front end engine may be embodied together with or separately from the back end server 502. In either case, processes performed at the front end engine 501 and the back end server 502 may be written onto a computer readable media and executed on a processor of any computing device. A user generated query is received 503 at the front end engine 501 and passed at 504 to the back end server 502. In one embodiment, the search query term may be stored in a database 505 associated with the back end server 502. The back end server 502 generates at least one facet relevant to the search query term using any appropriate computing method including random walk and dictionary based algorithms. The generated facets are returned at 506 to the front end engine 501, which renders these facets of the search query term into a TOC display 507. The user may then select a particular facet he or she wishes to explore further. The selected facet is sent at 508 to the back end server 502. In order to maintain the state of the user search experience, the original search query term may be recalled from the database 505 and the selected facet and the search query term are used to generate a set of sub facets. The set of sub facets for the selected facet and the search query term are sent 509 and received by the front end engine 501, which renders the TOC 507. The user may then choose to explore sub facets related to another facet by selecting an alternate facet or choose to explore a specific sub facet of the currently selected facet. In one embodiment, the selected facet may also be stored in a database associated with the back end engine and at this point, when the selected sub facet 510 is sent by the front end engine, the selected facet and the search query term would both be recalled. Using the recalled selected facet and the recalled search query term as well as the selected sub facet, a set of search results refined based upon this criteria is generated by the back end engine 502. These search results are sent at 511 to the front end engine 501, which renders the search results in the TOC display 507. The user may explore the displayed search results or select another facet or sub facet to explore while the state of the original search query term is maintained.

[0042] FIG. 5B is a block diagram 520 illustrating an alternative embodiment of the present invention. This embodiment also includes a front end engine 521 and a back end server 522. The front end engine may be embodied together with or separately from the back end server 522. In either case, processes performed at the front end engine 521 and the back end server 522 may be written onto a computer readable media and executed on a processor of any computing device. A user generated query is received at 523 at the front

end engine 521 and passed at 524 to the back end server 522. The back end server 522 generates at least one facet relevant to the search query term using any appropriate computing method including random walk and dictionary based algorithms. The generated facets are returned at 525 to the front end engine 501, which renders these facets of the search query term into a TOC display 526. The user may then select a particular facet he or she wishes to explore further. In order to maintain the state of the user search experience, the original search query term sent with the selected facet 527 to the back end server 522 and the selected facet and the search query term are used to generate a set of sub facets. The set of sub facets for the selected facet and the search query term are sent at 528 and received by the front end engine 521, which renders the TOC 526. The user may then choose to explore sub facets related to another facet by selecting an alternate facet or choose to explore a specific sub facet of the currently selected facet. In one embodiment, the selected sub facet, the select facet, and the original search query term are sent at 529 to the back end engine 522 which maintains the state of the search query term. Using the selected facet and the search query term as well as the selected sub facet, a set of search results determined by this criteria is generated by the back end engine 522. These search results are sent at 530 to the front end engine 521, which renders the search results in the TOC display 526. The user may explore the displayed search results or select another facet or sub facet to explore while the state of the original search query term is maintained.

[0043] FIG. 6A-D are a series of illustrations depicting an exemplary embodiment of the present invention. In one embodiment shown in FIG. 6A, the user may be provided a graphical user interface 600 by a front end engine 401 of FIG. 4 via a web browser 601. Other embodiments of a user display are within the scope of this invention, for example an offline navigation display or a text only interface. The front end engine may provide a prompt 602 for a user to input a search query term 603. In the example shown in FIG. 6A, the search query term input by a user is "Michael Jackson." Once the user provides the search query term, the search query term is received by the front end engine 401 and sent to the back end server 402. At this point, the state of the search query term may be maintained, as one of many possible embodiments, by storage within a database associated with the back end server 402 or, in another embodiment, it may be stored by front end engine 401 and resent or recalled at a later time. Once the facets associated with the search query term "Michael Jackson" have been determined using any of the above mentioned methods, the facets are returned to the front end engine.

[0044] In FIG. 6B in interface 610, the TOC is rendered and shows the facets generated for the search query term "Michael Jackson 604." In this example, the facets selected include: IMAGES 605, LYRICS 606, VIDEOS 607, BIOGRAPHY 608, and NEWS 609. In one embodiment, the back end server may also provide a list of search results 611 for the search query term which are rendered in the interface 610 by the front end engine. The facets selected in one embodiment are dependent on the selection criteria of the back end server 402 of FIG. 4. If a user input a different search query term such as "Wolfgang Mozart," a person of ordinary skill would recognize that the facets generated for a search query term "Michael Jackson" would not necessarily be relevant. For example, a search query term for "Wolfgang Mozart" may produce facets such as: BIOGRAPHY, WORKS, MUSEUMS, FICTIONAL DEPICTIONS. The facets provide search query term specific categories such that a user may explore different aspects of a particular search query term without diverting from the original search query term.

[0045] In FIG. 6C in interface 620, a user may select the facet NEWS 621 to explore search results related to "Michael Jackson" and "NEWS." Thus the selection of the facet NEWS 621 is sent by the front end engine to the back end server 402 of FIG. 4. In one exemplary embodiment, the front end engine may send both "Michael Jackson" and "NEWS" to the back end engine. In another embodiment, the front end engine may send "NEWS" to the back end engine, which in turn recalls the search query term "Michael Jackson" from an associated database. A search engine may be employed by the back end server to produce a set of search results for the selected facet "NEWS" and the search query term "Michael Jackson." In this particular example, the search results may be a list of web links. This set of search results is returned to the front end engine, which renders the search results or facet results 622 into the TOC display. The user may then click on any of the links or continue browsing the facets 605-609 associated with the search query term "Michael Jackson."

[0046] Since the state of the original search query term is maintained, all of the facets 605-609 remain available to the user without having to repeat the input of the original search query term. In FIG. 6D in interface 630, if the user then selects another facet to explore, such as VIDEOS 631, the selected VIDEOS 631 facet is returned to the back end server and the original search query term is supplied as well. The search engine may produce a set of web links relevant to "Videos" and "Michael Jackson." These facet results 632 are rendered in the TOC display for the user to evaluate while the original search query term is still available.

[0047] FIG. 7A-D are a series of illustrations depicting an exemplary embodiment of the present invention. In one embodiment shown in FIG. 7A in display 700, the user may be provided an interface by a front end engine 501 of FIG. 5 via a web browser 701. Other embodiments of a user display are within the scope of this invention, for example an offline navigation display or a text only interface. The front end engine may provide a prompt 702 for a user to input a search query term 703. In the example shown in FIG. 7A, the search query term input by a user is "Michael Jackson." Once the user provides the search query term, the search query term is received by the front end engine 501 of FIG. 5 and sent to the back end server 502. At this point, the state of the search query term may be maintained, as one of many possible embodiments, by storage within a database associated with the back end server 502 or, in another embodiment, it may be stored in a buffer associated with the front end engine 501. Once the facets associated with the search query term "Michael Jackson" have been determined using any of the above mentioned methods, the facets are returned to the front end engine.

[0048] In FIG. 7B in display 710, the TOC is rendered 704 and shows the facets generated for the search query term "Michael Jackson." In this example, the facets generated include: IMAGES 705, LYRICS 706, VIDEOS 707, BIOGRAPHY 708, and NEWS 709. In one embodiment, the back end server may provide a list of search results 711 for the search query term which are received and rendered in the interface 710 by the front end server. The facets selected in one embodiment are dependent on the selection criteria of the back end server 502 of FIG. 5. If a user input a different search query term such as "Wolfgang Mozart," a person of ordinary skill would recognize that the facets generated for a search query term "Michael Jackson" would not necessarily be relevant. For example, a search query term for "Wolfgang Mozart" may produce facets such as: BIOGRAPHY, WORKS, MUSEUMS, FICTIONAL DEPICTIONS. The facets provide search query term specific categories such that a user may explore different aspects of a particular search query term without diverting from the original search query term.

[0049] In FIG. 7C in display 720, a user may select the facet VIDEOS 721 to explore sub facets related to "Michael Jackson" and "VIDEOS." Thus the selection of the facet VIDEOS 721 is sent by the front end engine 501 to the back end server 502 of FIG. 5. In one exemplary embodiment, the front end engine may send both "Michael Jackson" and "VIDEOS" to the back end engine. In another embodiment, the front end engine may send "VIDEOS" to the back end engine, which in turn recalls the search query term "Michael Jackson" from an associated database. At the back end server 502, the sub facet

generation is accomplished in a manner similar to facet generation using exemplary random walk and dictionary based algorithms. In this embodiment, sub facets for VIDEOS and Michael Jackson are generated and returned to the front end engine.

[0050] In FIG. 7D in display 730, an exemplary embodiment of the TOC including the generated sub facets is shown. Once the sub facets for "Michael Jackson" and "VIDEOS" are received by the front end engine 501 of FIG. 5, the sub facets are rendered into the TOC display. In the embodiment, sub facets for the search query term "Michael Jackson" and "VIDEOS" may include MUSIC VIDEOS 731, INTERVIEWS 732, CONCERT VIDEOS 733, NEWS VIDEO 734, and TRIBUTES 735. A person would recognize that this set of sub facets are unique to the search query term "Michael Jackson" and "VIDEOS." A sub facet for "Wolfgang Mozart" and "WORKS" would yield possible sub facets such as SYMPHONIES, OPERAS, CONCERTOS, SONATAS, and CHAMBER MUSIC. The user may then explore the sub facets displayed for the search query term "Michael Jackson" and "VIDEOS" or the user may select a new facet to trigger the generation of another set of sub facets. In an exemplary embodiment, a user selects a sub facet TRIBUTES 735. Thus the selection of the sub facet TRIBUTES 735 is sent by the front end engine to the back end server 502 of FIG. 5. In one exemplary embodiment, the front end engine may send "Michael Jackson" and "VIDEOS" and "TRIBUTES" to the back end engine. In another embodiment, the front end engine may send "TRIBUTES 735" to the back end engine, which in turn recalls the search query term "Michael Jackson" and the facet "VIDEOS" from an associated database. In either case, the back end server 502 generates a list of search results which are refined by the search query term, "Michael Jackson," the selected facet "VIDEOS," and the selected sub facet "TRIBUTES." These sub facet results are returned to the front end engine which renders these search results for the selected sub facet in the TOC display 736. The user may then choose to explore the sub facet web links 736 relevant to the sub facet "TRIBUTES" displayed 735 in the exemplary embodiment or select another sub facet relevant to the facet "VIDEOS" or select another facet relevant to the original search query term "Michael Jackson." Since the state of the search query term is maintained, the user may fully explore all facets and sub facets relevant to "Michael Jackson" in the exemplary embodiment without re-entering the original search criteria.

[0051] FIG. 8A-B are illustrations of an alternative embodiment of the present invention. In the exemplary embodiment, the facets generated for the search query term are augmented with a set of static facets which are displayed for any search query term. In one embodiment shown in FIG. 8A in display 800, the user may be provided an interface by a front end engine 401 of FIG. 4 via a web browser 801. Other embodiments of a user display are within the scope of this invention, for example an offline navigation display or a text only interface. The front end engine may provide a prompt 802 for a user to input a search query term 803: in the example shown in FIG. 8A, the search query term input by a user is "Michael Jackson." Once the user provides the search query term, the search query term is received by the front end engine 401 and sent to the back end server 402 of FIG. 4. At this point, the state of the search query term may be maintained, as one of many possible embodiments, by storage within a database associated with the back end server 402 or, in another embodiment, it may be stored in a buffer associated with the front end engine 401. Once the facets associated with the search query term "Michael Jackson" have been determined using any of the above mentioned methods, the facets are returned to the front end engine.

[0052] In FIG. 8B in interface 815, the TOC is rendered 804 and shows the facets generated for the search query term "Michael Jackson." In this example, the facets generated include: IMAGES 805, LYRICS 806, VIDEOS 807, BIOGRAPHY 808, and NEWS 809. The facets selected in one embodiment are dependent on the selection criteria of the back end server 402 of FIG. 4. If a user input a different search query term such as "London," a person of ordinary skill would recognize that the facets generated for a search query term "Michael Jackson" would not necessarily be relevant. For example, a search query term for "London" may produce facets such as: HISTORY, TOURISM, BUSINESSES, NEWS. The facets provide search query term specific categories such that a user may explore different aspects of a particular search query term without diverting from the original search query term, i.e. maintaining the state. In addition to the search query term specific facets 806-809, a set of static facets are provided by the front end engine 401. In the exemplary embodiment, the static facets are a set of chronological categories indicating results for the last HOUR 810, DAY 811, WEEK 812, MONTH 813, and YEAR 814. Certainly other static facets might be useful as well, such as document type facets such as refining results to ".DOC," ".PDF," ".MP3," and ".MPG," or source based facets for example narrowing results to only ".COM" URLS or ".GOV" URLS. The static facets are not generated for each search query term and are not search query term

specific but may be selected by the user to further refine the results for a search query term. In a similar manner as described above, a selected static facet and/or a selected facet are sent to the back end server which also accesses the original search query term and uses these criteria to produce a set of search results. These search results are returned to the front end engine which in turn renders the search results into the facet results portion of the TOC display. Similarly, sub facets may also be employed if a user selects a facet in conjunction with a static facet.

[0053] Many different arrangements of the various components depicted, as well as components not shown, are possible without departing from the scope of the claims below. Embodiments of our technology have been described with the intent to be illustrative rather than restrictive. Alternative embodiments will become apparent to readers of this disclosure after and because of reading it. Alternative means of implementing the aforementioned can be completed without departing from the scope of the claims below. Certain features and subcombinations are of utility and may be employed without reference to other features and subcombinations and are contemplated within the scope of the claims.

CLAIMS

The invention claimed is:

1. One or more memory devices having computer-executable instructions embodied thereon that, when executed, perform a method for providing a user interface for exploring facets relevant to a user search query term, wherein a state of the user search query is maintained, the method comprising:

receiving at least one search query term from a user;

sending the at least one search query term to a back-end server;

receiving relevant facets for the at least one search query term, wherein each of the relevant facets categorizes a set of topics relevant to the at least one search query term and allows the user to refine the one or more search query terms;

displaying the relevant facets for the at least one search query term;

receiving a selection of one of the relevant facets;

sending the selected relevant facet to the back-end server;

receiving a set of search results based on the at least one search query term and the selected relevant facet input as a refinement query, wherein the refinement query is executed to generate the set of search results;

displaying:

(1) the relevant facets for the at least one search query term, wherein the relevant facets are generated by the back-end server using the at least one search query term recalled by the back-end server; and

(2) the received set of search results.

2. The method of Claim 1, the method further comprising storing the at least one search query term in a database.

3. The method of Claim 1, wherein the relevant facets are displayed in a web browser.

4. The method of Claim 1, the method further comprising receiving and displaying a set of search results for the at least one search query term with the relevant facets.

5. The method of Claim 1, the method further comprising displaying at least one static facet, wherein the static facet allows the user to refine the at least one search query term.

6. The method of Claim 1, the method further comprising providing a user search query prompt.

7. One or more memory devices having computer-executable instructions embodied thereon that, when executed, perform a method for, providing a user interface for exploring facets and sub facets relevant to a user-generated query, wherein a state of the user-generated query is maintained, the method comprising:

receiving at least one search query term from a user;

sending the at least one search query term to a back-end server;

receiving relevant facets for the at least one search query term, wherein each of the relevant facets categorizes a set of topics relevant to the at least one search query term and allows the user to refine the one or more search query terms;

displaying the relevant facets for the at least one search query term;

receiving a selection of one of the relevant facets;

sending the selected relevant facet to the back end server;

receiving sub facets for the at least one search query term and the selected relevant facet, wherein each of the sub facets categorizes a set of sub topics relevant to the at least one search query term and the selected relevant facet and allows the user to further refine the one or more search query terms;

displaying the sub facets and the relevant facets for the at least one search query term;

receiving a selection of one of the sub facets;

sending the selected sub facet to the back-end server;

receiving a set of search results based on the at least one search query term, the selected relevant facet, and the selected sub facet input, as a refinement query, wherein the refinement query is executed to generate the set of search results; and

displaying:

(1) the relevant facets for the at least one search query term, wherein the relevant facets are generated by the back-end server using the at least one search query term recalled by the back-end server;

(2) the sub facets for the at least one search query term and the at least one search query, wherein the relevant facets are generated by the back-end server using the at least one search query term and the relevant facet recalled by the back-end server; and

(3) the received set of search results.

8. The method of Claim 7, the method further comprising storing the at least one search query term in a database.

9. The method of Claim 8, the method further comprising storing the selected relevant in a database.

10. The method of Claim 7, wherein the sub facets and the relevant facets are displayed in a web browser.

11. The method of Claim 7, the method further comprising receiving and displaying a set of search results for the at least one search query term with the at least one relevant facets.

12. The method of Claim 7, the method further comprising displaying at least one static facet, wherein the static facet allows the user to refine the at least one search query term.

13. A computerized system for providing a user interface for exploring facets relevant to a user search query, wherein a state of the user search query is maintained, the system comprising:

a back-end server configured for:

generating relevant facets for at least one search query term received from a user, wherein the facet categorizes a set of topics relevant to the at least one search query term;

generating relevant facets for at least one search query term using the at least one search query term recalled by a front-end engine;

generating a set of search results based on the at least one search query term and a selected relevant facet input as a refinement query, wherein the refinement query is executed to generate the set of search results; and

the front-end engine configured for:

receiving relevant facets for the at least one search query term;

sending the at least one search query term and the selected relevant facet to the back end server;

receiving the set of search results based on the at least one search query term and the selected relevant facet input as the refinement query, wherein the refinement query is executed to generate the set of search results;

maintaining the relevant facets for the at least one search query term, wherein the relevant facets are generated by the back-end server using the at least one search query term recalled by the front-end engine;

displaying:

(1) the relevant facets maintained for the at least one search query term; and

(2) the received set of search results.

14. The system of Claim 13, the system further comprising a database configured for storing the at least one search query term.

15. The system of Claim 13,

the front-end engine further configured for:

receiving a set of search results for the at least one search query term with the relevant facets from the back end server; and

displaying the set of search results for the at least one search query term with the relevant facets.

16. The system of Claim 13, the front end engine further configured for sending the selected relevant facet to the back-end server.

17. The system of Claim 13, the back-end server further configured for:

generating sub facets for the at least one search query term and the selected relevant facet, wherein each of the sub facets categorizes a set of sub topics relevant to the at least one search query term and the selected relevant facet and allows the user to further refine the one or more search query terms; and

generating sub facets for the at least one search query term and the at least one search query, and wherein the relevant facets are generated by the back-end server using the at least one search query term and the relevant facet recalled by the front-end engine; and

generating a set of search results based on the at least one search query term, the selected relevant facet, and a selected sub facet input as a sub facet refinement query, wherein the sub facet refinement query is executed to generate the set of search results.

18. The system of Claim 14, the front-end engine configured for:

receiving the sub facets for the at least one search query term and the selected relevant facet;

sending the at least one search query term, the selected relevant facet, and the selected sub facet to the back end server;

receiving a set of search results based on the at least one search query term, the selected relevant facet, and the selected sub facet input as a sub facet refinement query, wherein the refinement query is executed to generate the set of search results; and

displaying:

(1) the relevant facets for the at least one search query term, wherein the relevant facets are generated by the back-end server using the at least one search query term recalled by the front-end server;

(2) the sub facets for the at least one search query term and the at least one search query, and wherein the relevant facets are generated by the back-end server using the at least one search query term and the relevant facet recalled by the front-end server.

19. The system of Claim 13, wherein the front end engine further comprises a web browser.

20. The system of Claim 13, the front end engine further configured for displaying at least one static facet, wherein the static facet allows the user to refine the at least one search query term.