

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
20 March 2008 (20.03.2008)

(10) International Publication Number
WO 2008/033511 A2

(51) International Patent Classification: **Not classified**

(21) International Application Number:
PCT/US2007/020017

(22) International Filing Date:
14 September 2007 (14.09.2007)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/844,534 14 September 2006 (14.09.2006) US

(71) Applicants and

(72) Inventors: **ROSENOFF, Douglas, T.** [US/US]; 26007 Se 156th Street, Issaquah, WA 98027 (US). **CROWHURST, Christopher** [GB/US]; 1701 Whitehall Road, Whitehall, MD 21161 (US).

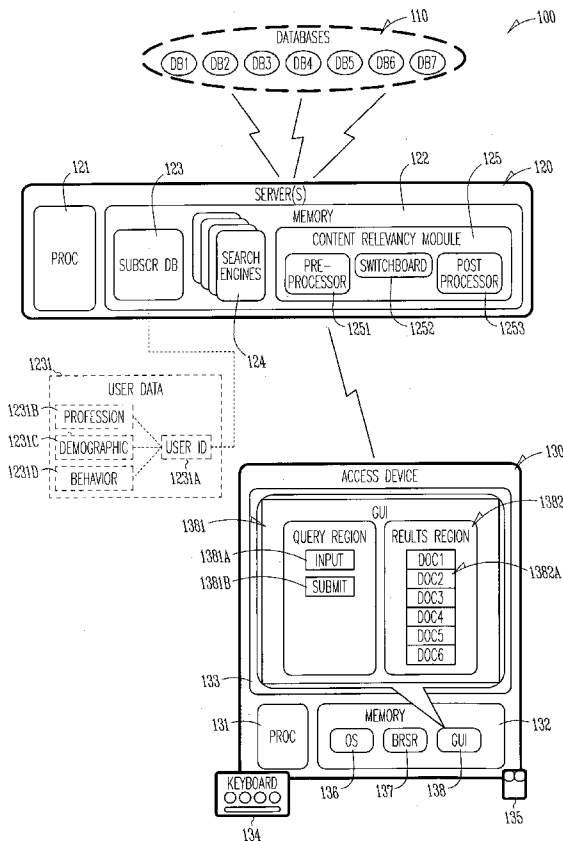
(74) Agents: **STEFFEY, Charles, E.** et al.; Schwegman Lundberg & Woessner, Pa, P.o. Box 2938, Minneapolis, MN 55402 (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— without international search report and to be republished upon receipt of that report

(54) Title: INFORMATION-RETRIEVAL SYSTEMS, METHODS, AND SOFTWARE WITH CONTENT RELEVANCY ENHANCEMENTS



(57) Abstract: The present inventors devised, among other things, systems, methods, and software for enhancing the relevancy of content presented to users in response to queries in an online information retrieval system. One exemplary system refines a user input query by making suggestions of alternatives queries, classifying the query using language processing, or selecting one or more databases or search engines as targets for the refined query. A switchboard module converts the refined query, administers one or more searches, and collects search results from one or more search engines based on the refined query. And, a post-processor module refines the collected search results by, for example, modifying the order of the results, removing inappropriate or undesirable content from the results, and /or applying historical performance analysis.

WO 2008/033511 A2

Information-Retrieval Systems, Methods, and Software with Content Relevancy Enhancements

5

Copyright Notice and Permission

A portion of this patent document contains material subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document or the patent disclosure, as it appears in the
10 Patent and Trademark Office patent files or records, but otherwise reserves all copyrights whatsoever. The following notice applies to this document:
Copyright © 2006, Thomson Global Resources, an entity organized under the laws of Ireland.

15

Cross-Reference to Related Application

This application claims priority to U.S. provisional application 60/844,534, which was filed on September 14, 2006 and which is incorporated herein by reference.

20

Technical Field

Various embodiments of the present invention concern information-retrieval systems and related query processing components and methods.

Background

25

The growth in popularity of the Internet and other computer networks has fueled not only an increasing availability, but an increasing appetite among computer users for digital information. Users typically seek access to this information using an access device, such a computer, to communicate with an online information- retrieval system. The information-retrieval system typically
30 includes a graphical user interface for entering and submitting requests for information, known as queries, to a remote search engine. The search engine identifies relevant information, typically in the form of documents, and returns a results list to the user's access device.

One problem that the present inventors recognized in conventional information-retrieval systems concerns users entering poorly defined queries and receiving search results that include hundreds or maybe even thousands of documents. Even though the documents are often ranked in terms of quantitative
5 relevance to the query, the task of identifying the documents most pertinent to the user's objective is onerous and time consuming. Moreover, if the query is poorly defined, the ultimate ranking of results is unlikely to be useful.

Accordingly, the inventors have identified a need to further improve how information-retrieval systems process user queries.

10

Summary

To address this and/or other needs, the present inventors devised, among other things, systems, methods, and software for enhancing the relevancy of content presented to users in response to queries in an online information
15 retrieval system. One exemplary system includes a database and a server for receiving and answering queries for content in the database. The server further includes a preprocessing module, a switchboard module, and a post-processing module. The preprocessing module refines a user input query by making suggestions of alternatives queries, classifying the query using language
20 processing, or selecting one or more portions of the legal-research database or search engines as a target for the refined query; The switchboard module converts the refined query and associated information developed by the pre-processor module, administers one or more searches, and collects search results based on the refined query. The post-processor module receives search results
25 based on the refined input query and refines the search results. Exemplary modifications include modifying the order of the search results, removing inappropriate or undesirable content from the results, and /or applying historical performance analysis to improve the relevance of the information presented to the user.

30

Brief Description of Drawings

Figure 1 is a block diagram of an exemplary information-retrieval system
100 which corresponds to one or more embodiments of the present invention.

Figure 2 is a flow chart of an exemplary method of operating system 100, which corresponds to one or more embodiments of the present invention.

Figure 3 is a conceptual diagram of a three-dimensional relevance data structure 300 that is used within system 100 and which corresponds to one or
5 more embodiments of the present invention.

Figure 4 is an alternative block diagram of various components of system 100, which highlights various data flows and which corresponds to one or more embodiments of the invention.

10 **Detailed Description of Exemplary Embodiment(s)**

This document describes one or more specific embodiments of an invention. These embodiments, offered not to limit but only to exemplify and teach the invention, are shown and described in sufficient detail to enable those skilled in the art to implement or practice the invention. Thus, where appropriate
15 to avoid obscuring the invention, the description may omit certain information known to those of skill in the art.

Exemplary Information-Retrieval System

20 Figure 1 shows an exemplary online information-retrieval system 100. System 100 includes one or more databases 110, one or more servers 120, and one or more access devices 130.

Exemplary Databases

25 Databases 110 includes a set of one or more databases. Exemplary legal databases include a caselaw database and a statutes database, which respectively include judicial opinions and statutes from one or more local, state, federal, and/or international jurisdictions. Exemplary legal databases also include legal classification databases and law reviews. Other exemplary databases provide
30 attorney, judge, law firm, product, and corporate profiles. In some embodiments, the caselaw documents are logically associated via a data structure with documents or profiles in other databases. Other embodiments may include non-legal databases that include financial, scientific, or health-care information. Still other embodiments provide public or private databases, such as those made

available through INFOTRAC. Some embodiments also include web sites and web pages.

Databases 110, which take the exemplary form of one or more electronic, magnetic, or optical data-storage devices, include or are otherwise associated with respective indices (not shown). Each of the indices includes terms and phrases in association with corresponding document addresses, identifiers, and other conventional information. Databases 110 are coupled or couplable via a wireless or wireline communications network, such as a local-, wide-, private-, or virtual-private network, to server 120.

10

Exemplary Server

Server 120, which is generally representative of one or more servers for serving data in the form of webpages or other markup language forms with associated applets, ActiveX controls, remote-invocation objects, or other related software and data structures to service clients of various “thicknesses.” More particularly, server 120 includes a processor module 121, a memory module 122, a subscriber database 123, search modules or engines 124, and a content-relevancy module 125.

Processor module 121 includes one or more local or distributed processors, controllers, or virtual machines. In the exemplary embodiment, processor module 121 assumes any convenient or desirable form.

Memory module 122, which takes the exemplary form of one or more electronic, magnetic, or optical data-storage devices, stores subscriber database 123, primary search module 124, secondary search module 125, and search enhancement module 126.

Subscriber database 123 includes subscriber-related data for controlling, administering, and managing pay-as-you-go or subscription-based access of databases 110. In the exemplary embodiment, subscriber database 123 includes one or more preference data structures, of which data structure 1231 is representative. Data structure 1221 includes a customer or user identifier portion 1231A, which is logically associated with one or more user data fields or preferences, such as fields or preferences 1231B, 1231C, and 1231D. Field 1231B includes data regarding the profession of the user. Field 1231C includes demographic data regarding the user. Field 1231D includes data regarding

online behavior of the user. (In the absence of a temporary user override, for example, an override during a particular query or session, the default value for the user preference governs.)

Search module 124 includes one or more search engines and related user-
5 interface components, for processing queries against one or more of databases 110. In some embodiments, one or more search engines associated with search module 124 provide Boolean, tf-idf, natural-language search capabilities, or search engines specialized for searching multi-media or types of content. In the exemplary embodiment, search engines are regarded as third-party, commodity
10 black-box components or environments that are selected according to the desires (explicit or derived) of the user and/or requesting application. In the above figure, the search engines are coupled directly or operatively to one or more corresponding databases (not shown). In some embodiments, two or more of the search engines are coupled to the same database; whereas in other embodiments,
15 the search engines are each coupled to separate and independent databases. In some instances, the databases are under control or ownership of third-parties.

Content-relevancy module 125 includes machine readable and/or executable instruction sets for enhancing the relevancy of content provided to users in response to their queries. To this end, content-relevancy module 125
20 includes a pre-processor module 1251, a switchboard module 1252, and a post-processor module 1253.

In general, the function pre-processor module 1251 is to process a user input query, refine it, make suggestions of alternatives, classify the query using language processing, and select the appropriate content sources or search
25 engines. Switchboard module 1252 converts the refined query and associated information developed by pre-processor module 1251 to administer one or more searches, and collect the search results. Post-processor 1253 refines the search results, modifying the order of results, removing inappropriate or undesirable content, and/or applying historical performance analysis to improve the
30 relevance of the information ultimately presented to a user of an access device, such as access device 130. (More detailed description of the function and features of the modules within content-relevance module 125 is provided below.)

Exemplary Access Device

Access device 130 is generally representative of one or more access devices. In the exemplary embodiment, access device 130 takes the form of a personal computer, workstation, personal digital assistant, mobile telephone, or
5 any other device capable of providing an effective user interface with a server or database. Specifically, access device 130 includes a processor module 131 one or more processors (or processing circuits) 131, a memory 132, a display 133, a keyboard 134, and a graphical pointer or selector 135.

Processor module 131 includes one or more processors, processing
10 circuits, or controllers. In the exemplary embodiment, processor module 131 takes any convenient or desirable form. Coupled to processor module 131 is memory 132.

Memory 132 stores code (machine-readable or executable instructions) for an operating system 136, a browser 137, and a graphical user interface
15 (GUI)138. In the exemplary embodiment, operating system 136 takes the form of a version of the Microsoft Windows operating system, and browser 137 takes the form of a version of Microsoft Internet Explorer. Operating system 136 and browser 137 not only receive inputs from keyboard 134 and selector 135, but also support rendering of GUI 138 on display 133. Upon rendering, GUI 138
20 presents data in association with one or more interactive control features (or user-interface elements). (The exemplary embodiment defines one or more portions of interface 138 using applets or other programmatic objects or structures from server 120 to implement the interfaces shown above or described elsewhere in this description.)

In the exemplary embodiment, each of these control features takes the
25 form of a hyperlink or other browser-compatible command input, and provides access to and control of a query region 1381 and a search-results region 1382. User selection of the control features in region 1382, specifically input of a textual query into input field 1381A and submission of the query to server 120
30 via actuation of submit button 1381B, results in presentation of search results list 1382A in results region 1382. Selection of a listed document from list 1382A results in retrieval of and display of at least a portion of the corresponding document within a region of interface 138 (not shown in this figure.) Although

Figure 1 shows region 1381 and 1382 as being simultaneously displayed, some embodiments present them at separate times.

Exemplary Method(s) of Operation

5

Figure 2 shows a flow chart 200 of one or more exemplary methods of operating a system, such as system 100. Flow chart 200 includes blocks 210-250, which are arranged and described in a serial execution sequence in the exemplary embodiment. However, other embodiments execute two or more
10 blocks in parallel using multiple processors or processor-like devices or a single processor organized as two or more virtual machines or sub processors. Other embodiments also alter the process sequence or provide different functional partitions to achieve analogous results. For example, some embodiments may alter the client-server allocation of functions, such that functions shown and
15 described on the server side are implemented in whole or in part on the client side, and vice versa. Moreover, still other embodiments implement the blocks as two or more interconnected hardware modules with related control and data signals communicated between and through the modules. Thus, the exemplary process flow applies to software, hardware, and firmware implementations.

20 In block 210 the method begins with receiving a query from a user. In the exemplary embodiment, this entails a user using a browser capability in access device 130 to access online information-retrieval system 100, specifically server 120 using a conventional login process. Once logged in, a user interface, such as interface 138 is rendered, enabling the user to define and submit a query
25 to server 120. Execution proceeds to block 220.

Block 220 entails pre-processing the received query along with related user information. In the exemplary embodiment, pre-processing of the query occurs within pre-processor module 1251 of content-relevancy module 125 in server 120. However, in some embodiment, pre-processor module may be
30 located in whole or in part within access device 130 or within a separate server. Pre-processing entails parsing the query and related information to separate out specific query term syntactic elements for additional processing. This additional processing can include one or more of the following:

- Selecting of one or more ontologies or taxonomies based on the provenance of the requesting application (q.v. selection of the KeySearch for general predefined queries from an online legal research system such as Westlaw, selection of West Legal Directory for queries coded with a field of attorney or judge, or selection of West Medical Taxonomy for queries coming from a malpractice tab of an online research system).
- Expanding and/or refining the query to include appropriate broader content topics or concepts as well as more specific topics or query terms. For example, assume a query of the single term “maple”. Via the syntactic relationships derived from the taxonomies and other data structures, additional query terms such as “Acer”, “Maple Sugar”, “Maple Leaf”, “Hockey” and “Maple Leaf Hockey Team” “Toronto Sports Teams” might be added depending on the taxonomy(s) selected in a previous step.
- Expanding and refining the query to change or select various content sources or search engines by associating the query terms with information found in the taxonomies or ontologies. For example, a query including the term SBX might automatically assign the query to a search engine with better financial coverage or to a SEC filing content source rather than a general purpose search engine. In some embodiments, pre-processor module 1251 uses the information in the query and, optionally, additional information (for example, user or workflow context information) passed by the requesting client application, refines the query. Exemplary refinement includes tailoring the query to the profession, demography, behavior, and/or other defining characteristic of the original user (as indicated in the context information or as may be associated with a user access credentials for the information-retrieval system.). Pre-processor module 1251 also selects one or more appropriate search engines based on an analysis of the information held in the refined query and its related data.
- Correcting the query terms for spelling, case, and syntax, depending on both dictionary lookup information and the taxonomy or other

semantic lookup method used (q.v. “Aser Palmatum” is corrected to “Acer palmatum” if an taxonomy associated with tree pathology is selected in a prior step or converted to the single query term “Ace” if an ontology associated with gaming is selected.)

5 To support these activities, pre-processor module 1251 includes a variety of taxonomies, ontologies, folksonomies, synonym lists, reserved or restricted term lists, stop word lists classification algorithms, and semantic tools for processing received queries. Data structures for the taxonomies, ontologies, and so forth may be editorially or programmatically populated and may consist of
10 proprietary and/or public domain data. These structures may be shared among all clients accessing pre-processor module 1251 or restricted to specific requesting applications or clients. These structures may be fixed in that they are not subject to feedback from other components or they may be adaptive in that the contribution they provide for an algorithmic choice may change based on user
15 feedback or input. Each of the various data structures have one or more algorithms for associating specific query terms or semantic elements to the entries in the various data structures. Such relationships maybe combined to produce stronger or weaker recommendations for any of the methods of refining the query mentioned above.

20 Pre-processor module 1251 also includes functionality for sending clarification and feedback messages back to the user prior to executing the query. Such messages for relevance clarification allow the user to refine their query before incurring the expense of actually running one or more searches. For example, in response to a query including the term Salvia, the pre-processor
25 module can request the user for clarification of whether he is interested in “Hallucinogenic Substances” (based on a related concept entry for “Salvia Divinorum” in a legal taxonomy). Such clarification messaging are made to the requesting application with the notion that the resulting clarifications (if any) serve as grist for recursive processing of the query before passing it and any
30 related data over to the Switchboard component for forwarding to appropriate search engines.

Pre-processor module 1251 keeps a record of selected operations performed on the input query and the refined query as well as a record of taxonomies, content sources, and/or search engines selected, passing the

information to the Switchboard component. Pre-processor module 1251s also provides a mechanism for updating adaptive taxonomies and similar semantic relationship data structures by receiving feedback from the Post-Processor. Such feedback may be used to dynamically alter the weights of any given factor used
5 to select related terms, refine concepts or queries, or select content sets or search engines. After pre-processing of the query, execution continues at block 230.

Block 230 entails collecting search results based on the refined or preprocessed query. In the exemplary embodiment, this entails the pre-processor module forwarding the refined or pre-processed query and related information to
10 switchboard module 1252. In turn, switchboard module 1252 forwards the refined query to one or more search engines identified by preprocessor module 1251. (In the exemplary embodiment the switchboard module and the pre-process are implemented using the same server; however, in other embodiments they may be distributed.)

15 More specifically, the exemplary switchboard module performs several functions with the preprocessed query and related information as input:

- Converting the refined query and related data to the syntax required for one or more search engines;
- Submitting the syntactically correct refined query to one or more
20 search engines;
- Caching the query and related data, including search engine selection analysis and parameterization;
- Waiting for query results to return from one or more search engines according to a set of defined parameters;
- 25 • Gathering the search results from one or search engines
- Organizing and/or federating the results from one or more search engines
- Passing the search engine results and the related data to post-processor module 1253.

30 Switchboard module 1252 also provides a feedback channel for taking information from the post-processor and returning it to pre-processor module 1251. Such feedback data is used to continually refine and enhance the functionality of pre-processor module 1251 by tuning the taxonomies,

ontologies, classification schemes, and semantic analysis techniques used in refinement of later queries. Execution continues at block 240.

Block 240 entails post-processing of the search results. To this end in the exemplary embodiment, post-processor 1253 takes as input one or more search results, as well as the refined query and its related information. It then performs a variety of taxonomic, semantic, syntactic, classification, and combinatorial operations on the search result(s), eventually creating a new, highly refined result. The refined result, possessing significantly more targeted, relevant results for the user, is then passed back to the requesting application which then passes it back to the user.

To accomplish this, the post-processor module may perform a number of operations including:

- Federating the results from one or more search engines using a variety of algorithmic and/or user interaction methods;
- Formatting the results according to a generic template or one provided by the requesting application;
- Collecting and holding feedback information provided by the requesting application
- Refining its own taxonomic, semantic, syntactic, classification and combinatorial operations based on the feedback;
- Returning the feedback information to the pre-processor module via the switchboard module for eventual refinement of pre-processing operations;
- Logging the execution of the query, the refinements made to the results, and the result itself;
- Monitoring and recording the reception and disposition of feedback from the requesting application.

To facilitate reordering and/or redacting or more generally refining the collected search results, post-processor module 1253 also includes a series of data structures. These data structures, which are stored in a memory of or otherwise associated with the post-processor module, may include various white lists (web sites of known high relevance), black lists (sites to be redacted from result lists), as well as data structures and methods for relating specific terms or topics to particular domains, sub-domains, pages, and/or content types. Such

relationships between topics and sites on the Internet (characterized by a Uniform Resource Name) maybe created editorially, programmatically by deriving them from one or more ontologies, or taxonomies, and/or adaptively from data provided by user interaction.

5 Figure 3 shows one exemplary implementation of three-dimensional data structure, referred to herein as a relevance cube 300. Cube 300 is divided into cells with the x-value of any specific cell in the cube providing a list of URL locations, the y-axis consisting of one or more concepts or terms associated with the relevance and the content of the cube include some weight value.

10 A field of cells may be defined by deriving the terms from a taxonomy to provide the list of terms along the y-axis of the stack and a list of “white” (highly relevant) sites along the x-axis. Several fields of cells maybe combined to form a preference cube, associated with a specific user, for example in a user data structure stored in subscriber database 123. When the post-processor analyzes a
15 raw result set returned from a search engine, an algorithm breaks it down into individual search results associated with a particular result location to which it points, a title, relevance value, and perhaps a snippet of surrounding text..

 One embodiment provides another algorithm to scan the individual search results for the location to which it points. If one of the locations is found,
20 the query terms, the page title, and/or any terms in the individual result snippet set are compared with the topic terms to produce one or more derived terms. If an intersection for the location along the x-axis and the derived terms along the y-axis exists, the weight in the corresponding cell is looked up and then be multiplied with a conventional relevance score for the particular result. The
25 newly weighted scores can then be used to bubble the individual search result higher or lower to produce the new refined search result.

 Fields may be combined to produce cubes. By traversing along the z-axis through the cube while holding the x-and y-values constant, a combined weight can be derived for a particular topic/location combination. Such a combined
30 weight can be used to reflect numerous additional criteria at the same time.

 In the example of Figure 3, four fields of cells are shown: editorial, click-through, social, and user. The editorial field is created editorially by deriving the topics from a fixed taxonomy and combining those with selected locations on the World Wide Web. Its weighting values are static in the sense that are not subject

to change by user feedback. However the entire field may grow or shrink when additional topics are added to the topology or additional locations are added.

The click-through field is a derived field. It initially consists of no weights but is populated when an individual user clicks through from a search result into a particular document page. Such a click through evidences some decision by a particular user that this page has a higher value than others in a particular search result and hence deserves a higher weighting when subsequent searches results are created by other users. The click through is fed back from the access device to the server and ultimately to the post-processor module. The net of this activity is to raise or lower the weight of a particular cell in this field.

The social field is a derived field. Like the click-through field it is initially blank but populated when users specifically give feedback that the result set is particularly relevant to the terms in the refined query. In such cases, the weights in multiple topic/location cells are raised or lowered to express the user preference.

The user field is a derived field. Like other derived fields it is initially blank but can be populated by the user activity in a variety of ways. For example a particular customer firm or user may block out lists or sites so that they are not included in refined results at all (q.v. black list the sites), raise the value of certain sites as highly relevant to them (q.v. white list the site), and block out or enhance the weights of various topic terms. In such cases the resulting field may be made highly specific to a particular customer, company, or even requesting application.

Note that there is no limiting factor on the number of criteria that may be expressed as a particular field. A relevance field may consist of an arbitrary number of topics and locations. Similarly a relevance cube may consist of N arbitrary fields. Note also that different requesting applications may share one or more relevance cubes. Similarly a particular application may restrict itself to a one or more selected cubes, some of which may be proprietary to it alone.

After post-processing of the search results in block 240 to produce a refined set of search results, execution continues at block 250. Block 250 entails presenting the refined search results to the use via a graphical user interface such as interface 138 of access device 130 in Figure 1. As noted above, user

interaction with the search results is ultimately fed back to post-processor to enable refinement of relevance weights.

Exemplary Data Flow

5 Figure 4 shows an alternative block diagram of a portion of system 100, highlighting various data flows between components of content-relevancy module 125, search engines 124, and access device 130. Notably, Figure 4 shows flow of a raw query 410 from access device 130 to pre-processor module 1251, and a refined query 420 from pre-processor module 1251 to switchboard
10 module 1252. Switchboard module 1252 communicates the refined query and receives search results from search engines 124, aggregating and passing the received results in the form of raw search results 430 to post-processor module 1253. After post-processing, post-processor module 1253 communicates the results as relevant search results 440 to access device 130. Additional data
15 flows, in the exemplary system, include query feedback data 450 from access device 130 to pre-process module 1251 through switchboard module 1252 to post-processor module 1253. Result feedback data 460 flows from access device 130 to post-processor module 1253, facilitating adjustment of weights within relevance cubes 1253A.

20

Conclusion

The embodiments described above are intended only to illustrate and teach one or more ways of practicing or implementing the present invention, not to restrict its breadth or scope. The actual scope of the invention, which
25 embraces all ways of practicing or implementing the teachings of the invention, is defined only by the issued claims and their equivalents.

CLAIMS

What is claimed is:

1. A system comprising:
a legal-research database including caselaw documents and statutory documents; a server operatively coupled to the legal-research database for receiving and answering queries for content for browser-equipped client access devices, wherein the server is further configured with:
means for pre-processing or refining a user input query by make suggestions of alternatives queries, classifying the query using language processing, or selecting one or more portions of the legal-research database or one or more search engines to receive the refined query;
switchboard means for converting the refined query and associated information developed by the pre-processor means, administering searches and collecting search results from search engines based on the refined query;
means for post-processing or refining the collected search results based on the refined input query, including means for modifying the order of results, means for removing inappropriate or undesirable content from the results, or means for applying historical performance analysis in presenting search results to users.
2. The system of claim 1, wherein the legal-research database include legal brief documents and law-review documents;
3. The system of claim 1, wherein the means for pre-processing includes a legal topic classification system.
4. The system of claim 1, wherein the means for post-processing or refining the collected search results includes a multi-dimensional data structure relating topics in a taxonomy to Uniform Resource Locators and associated relevance weighting factors.

5. A method of operating an online legal research system, the method comprising:
receiving a query for legal research information from a client access device communicating via a computer network;
obtaining search results based on the query, with the search results having a first ordering based on a first relevance ranking;
automatically modifying the search results based on predetermined relevance criteria; and
communicating the modified search results to the client access device.
6. The method of claim 5, wherein obtaining search results based on the query includes refining the query by prompting a user of the client access device to respond to one or more questions, wherein the search results are based on the refined query.
7. The method of claim 5, wherein automatically modifying the search results includes excluding one more items from the search results based on a predetermined list of Uniform Resource Locators.
8. The method of claim 5, wherein automatically modifying the search results includes adding one more items from the search results based on a predetermined list of Uniform Resource Locators.
9. The method of claim 5, wherein automatically modifying the search results includes reordering one or more items in the list based on predetermined relevance criteria.
10. The method of claim 5, wherein the predetermined relevance criteria are based in part on historical usage statistics associated with one or more Uniform Resource Locators.

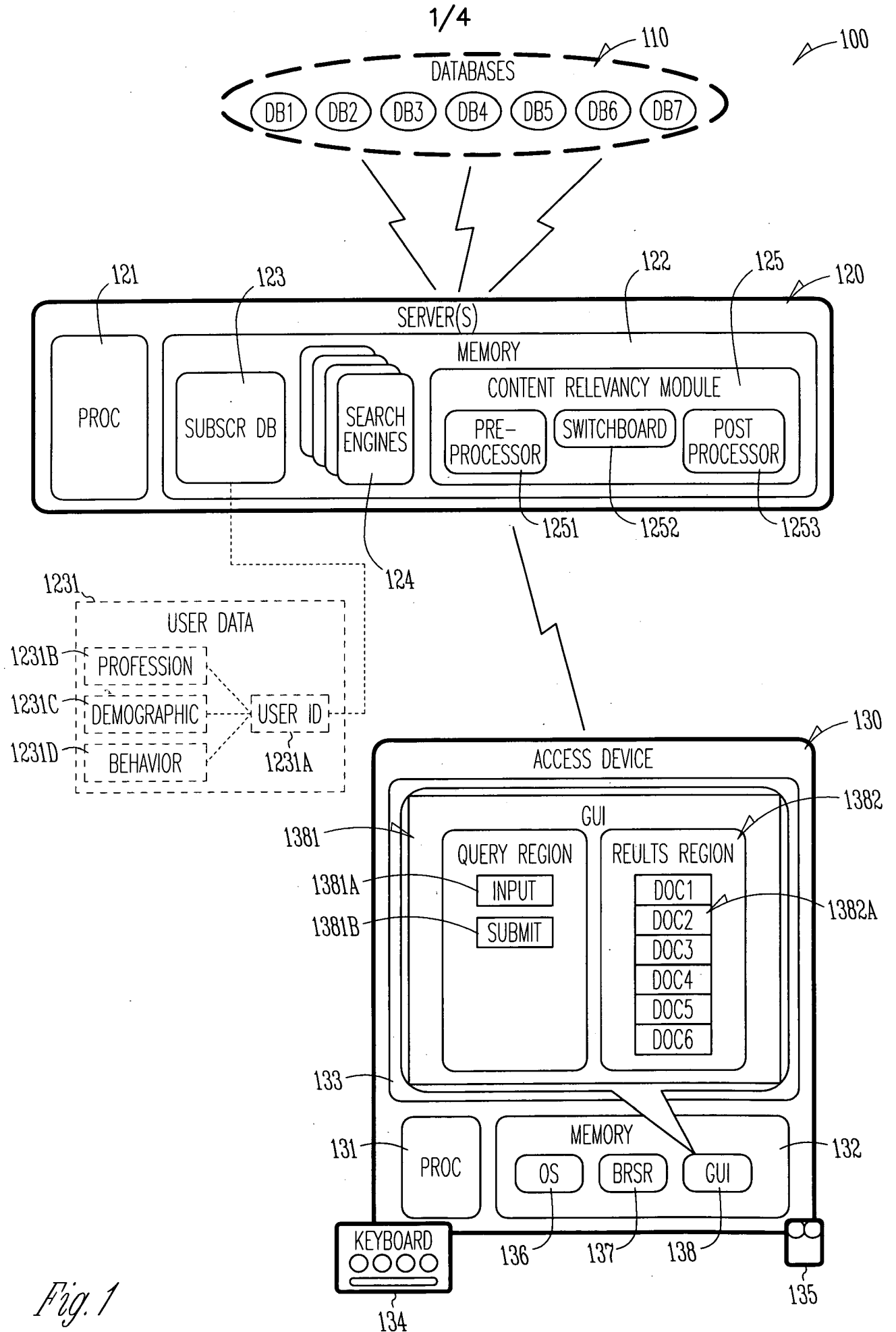


Fig. 1

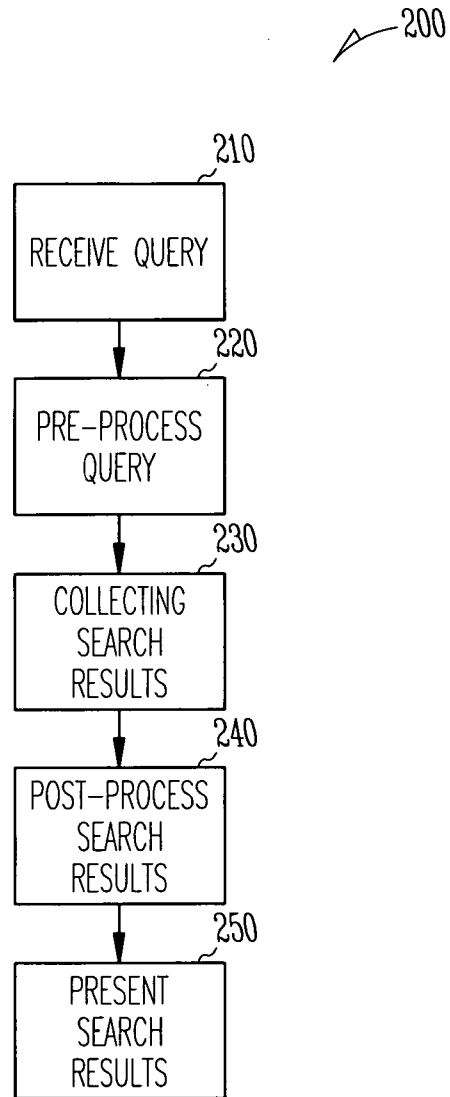


Fig. 2

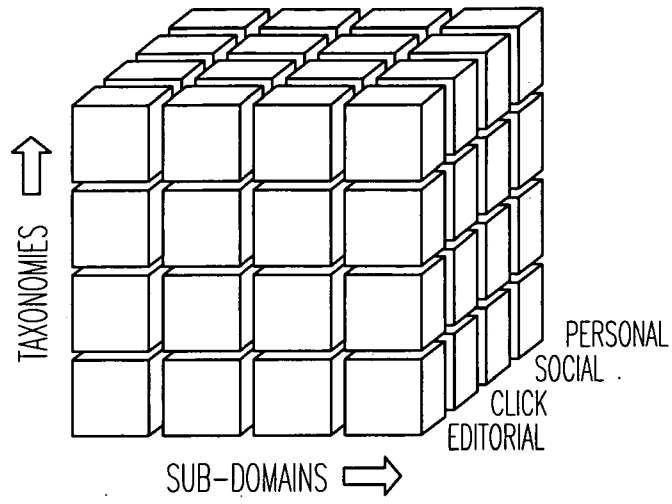


Fig. 3

Fig. 4

