

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
31 January 2008 (31.01.2008)

PCT

(10) International Publication Number
WO 2008/014262 A1

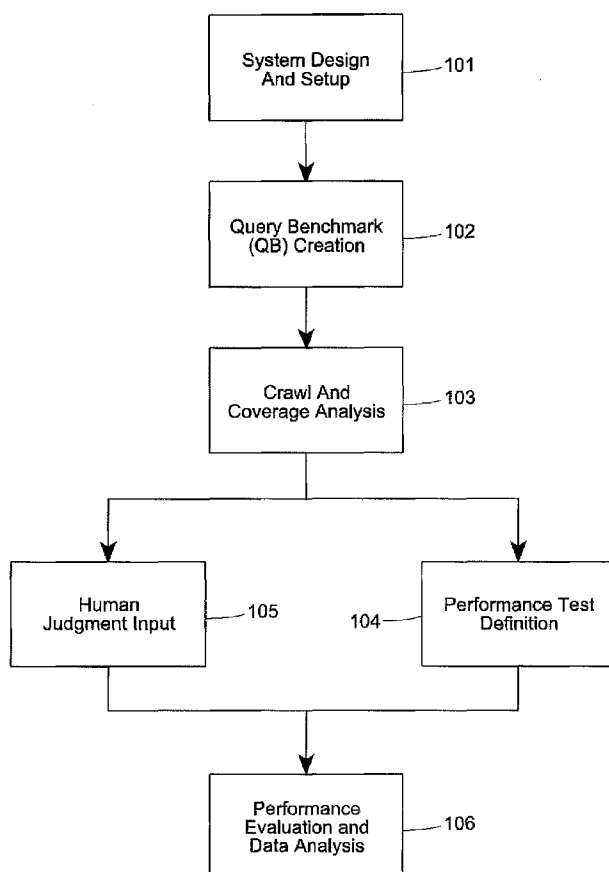
- (51) International Patent Classification:
G06Q 99/00 (2006.01) *G06F 17/30* (2006.01)
- (21) International Application Number:
PCT/US2007/074224
- (22) International Filing Date: 24 July 2007 (24.07.2007)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
11/493,268 25 July 2006 (25.07.2006) US
- (71) Applicant (for all designated States except US): **YAHOO! INC.** [US/US]; 701 First Avenue, Sunnyvale, CA 94089 (US).
- (72) Inventors; and
- (75) Inventors/Applicants (for US only): **CHANG, Chi-chao** [US/US]; 3313 Duke Court, Santa Clara, CA 95051 (US). **CHEN, Jian** [US/US]; 34313 Larkspur Court, Union City, CA 94587 (US).
- (74) Agent: **DECARLO, James, J.**; Greenberg Traurig, LLP, 200 Park Avenue, 34th Fl., New York, NY 10166 (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).

[Continued on next page]

(54) Title: SYSTEM AND METHOD OF INFORMATION RETRIEVAL ENGINE EVALUATION USING HUMAN JUDGMENT INPUT



(57) Abstract: An information retrieval engine evaluation system and method is disclosed, which uses judgment input, or feedback, received from one or more individuals, or judges. Judgment input is provided by the one or judges, each of whom review at least one aspect of performance of a software application, and provide judgment input in the form of responses to questions. The judgment input is received and analyzed, and can be used to generate the one or more metrics. The metrics can be examined to evaluate at least one indicator in order to determine performance of the software application.

WO 2008/014262 A1

**Published:**

- *with international search report*
- *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

SYSTEM AND METHOD OF INFORMATION RETRIEVAL ENGINE EVALUATION USING HUMAN JUDGMENT INPUT

BACKGROUND

[0001] 1. Field

[0002] This disclosure relates to computing system performance, and more particularly to a system and method for defining and measuring information retrieval performance using a software application (e.g., a search engine), or portion thereof, using one or more indicators of performance.

[0003] 2. General Background

[0004] In a case that an existing software application is modified, or a new software application is created, it would be beneficial to be able to be able to measure the performance of the application.

[0005] One example of such a software application, which has become an invaluable tool to search a document store and retrieve information from the data store, is a search engine. With the advent of computer networks, including the World Wide Web or Internet, which have facilitated and expanded access to such data stores, a search engine has become a tool which is used on an everyday basis.

[0006] Typically, a search engine has functionality to search and index available information. A software agent, typically referred to as crawler, can be used to traverse the computer network, to locate and identify data items available via computer network. Typically, the search engine uses one or more indices, each of which associates a data item (e.g., a document) available on the computer network and at least one keyword, which corresponds to contents of the data item. In response to a search request, the search engine searches one or more indices to identify documents based on criteria specified in a search request. The search engine typically ranks the result items identified in the search. For example, the search results can be ordered based on one or more criteria, such as relevance, date, etc.

[0007] Modifying a search engine's functionality, including any of the areas discussed above, can impact the search engine's performance, e.g., can result in a change in the content and/or appearance of the search results. It would be beneficial to be able to measure what, if any, impact a modification in a software application, such as a search engine, has on the application's performance. In addition, it would be beneficial to be able to design one or more tests, or experiments, to measure such an impact. Further still, it would be beneficial to be able to measure a user's perception of, and/or user impact with respect to, such a modification.

SUMMARY

[0008] The present disclosure seeks to address these failings and to provide a method and system for measuring information retrieval software application performance.

[0009] Embodiments of the present disclosure can be used to evaluate at least one indicator, comprising one or more metrics, to determine a degree to which a change impacts performance of a search engine. One example of an indicator is search result relevance. Other examples of indicators include, without limitation, distance, and coverage. Embodiments of the present disclosure determine one or more metrics which operate on judgment input, or feedback, provided by one or more individuals, or judges. Judgment input is provided by one or judges, each of whom review search results generated by the search engine and provide judgment input in the form of responses to questions posed to the judge concerning the search results. The judgment input is received and analyzed, and can be used to generate the one or more metrics. The metrics can be examined to evaluate at least one indicator, e.g., relevance, in order to determine the degree to which a change impacts performance of the search engine.

[0010] In accordance with one or more embodiments, an information retrieval engine evaluation is provided, which comprises identifying a query benchmark comprising a plurality of queries, the queries having corresponding search results, obtaining judgment input from one or more judges, the judgment input corresponding to the set of search results, determining, using the judgment input obtained from the one or more judges, at least one metric corresponding to an indicator of performance, a first value of the at least one metric corresponding to a first information retrieval engine and a second value of the at least one metric corresponding to a second information retrieval engine, and comparing the first and

second values of the at least one metric to evaluate the performance indicator so as to evaluate performance of the first information retrieval engine relative to the second information retrieval engine.

[0011] In accordance with one or more embodiments, search engine performance measurement using a set of stored queries, results and judgments is provided, which comprises generating a query benchmark from the set of stored queries, the query benchmark includes one or more queries, obtaining query results using the one or more benchmark queries, retrieving one or more stored results associated with the plurality of queries, retrieving a judgment associated with at least one stored result, predicting a judgment associated with one or more of the obtained results based on the at least one stored result, and determining a performance measure using the retrieved and predicted judgments.

DRAWINGS

[0012] The above-mentioned features and objects of the present disclosure will become more apparent with reference to the following description taken in conjunction with the accompanying drawings wherein like reference numerals denote like elements and in which:

[0013] Figure 1 provides a process overview for use with one or more embodiments of the present disclosure.

[0014] Figure 2 provides an example of an architecture overview comprising components for use in performance evaluation in accordance with one or more embodiments of the present disclosure.

[0015] Figure 3 provides an example of a portion of a judgment input user interface for use in accordance with one or more embodiments of the present disclosure.

[0016] Figure 4A provides a Spearman footrule distance example in accordance with one or more embodiments of the present disclosure.

[0017] Figure 4B provides a Kendall Tau distance determination example in accordance with one or more embodiments of the present disclosure.

[0018] Figure 5, which comprises Figures 5A to 5D, provides various output generated in accordance with one or more embodiments of the present disclosure.

[0019] Figure 6 provides an example of a database structure of analysis database 210 in accordance with one or more embodiments of the present disclosure.

[0020] Figure 7 provides an example of output from a Rasch utility, configured to operate on a model in accordance with one or more embodiments of the present disclosure.

DETAILED DESCRIPTION

[0021] In general, the present disclosure includes a system and method for defining and measuring performance of a software application (e.g., a search engine), or portion thereof, using one or more indicators of performance.

[0022] Certain embodiments of the present disclosure will now be discussed with reference to the aforementioned figures, wherein like reference numerals refer to like components.

[0023] Embodiments of the present disclosure can be used to evaluate at least one indicator, comprising one or more metrics, to determine a degree to which a change impacts performance of one or more search engines. One example of an indicator is search result relevance. Other examples of indicators include, without limitation, distance, and coverage. Embodiments of the present disclosure determine one or more metrics which operate on judgment input, or feedback, provided by one or more individuals, or judges. Judgment input is provided by one or judges, each of whom review search results generated by the search engine and provide judgment input in the form of responses to questions posed to the judge concerning the search results.

[0024] In accordance with one or more embodiments presently disclosed, a mechanism can be used to predict judgment input as an enhancement, or in an absence of, human judgment input. For example, an artificial intelligence engine can be used to predict judgment input received from human judges. The artificial intelligence engine has knowledge of prior human judgment input so as to determine a manner in which a human judge would likely respond to a given search result, or set of search results. In other words, the artificial intelligence engine can be educated based on prior human judgment input in order to predict human judgment input. An “artificial intelligence judge” educated in this manner can then be used as a substitute for, or in addition to, human judges, and can provide judgment input, in accordance with one or more embodiments of the present disclosure.

[0025] In addition, a result for which there is no judgment input (e.g., a result that has not been judged to be either relevant or nonrelevant) can be given a judgment prediction. For example, in accordance with one or more embodiments, a result without corresponding judgment input can be predicted to be relevant, or alternatively can be predicted to be nonrelevant. A prediction can be further refined based on the ranking of the result in a search result set in accordance with one or more embodiments. For example, a result which is one of the highest ranking results in a search request (e.g., in the top five results returned by a search engine) can be predicted to be relevant, wherein a result which is one of the lowest ranking results can be predicted to be nonrelevant, or vice versa.

[0026] In accordance with one or more embodiments, judgment input is received and analyzed, and can be used to generate the one or more metrics. The metrics can be examined to evaluate at least one indicator, e.g., relevance, in order to determine the degree to which a change impacts performance of the search engine.

[0027] Embodiments of the present disclosure are described with reference to search engine performance testing. However, it should be apparent that disclosed embodiments are not limited to search engines, but are applicable to performance testing of any type of software application, consumer electronic device, computing device, etc., the performance of which can be evaluated using human judgment input.

[0028] Figure 1 provides a process overview for use with one or more embodiments of the present disclosure. Step 101 comprises system design and setup, which can comprise identification of a methodology, and general guidelines, for use in evaluating a search engine. For example, system design and setup can be used to define an experimental framework that can be used to evaluate search engine performance, such as establishing guidelines for use in defining a query benchmark, or "QB", (e.g., number of queries used in a QB, a number of results to be judged for each query in a QB, a criteria for selecting queries, etc.). In a case of selecting queries for a query benchmark, a selection can be automatic, or can be specified by a user, such as a software developer or engineer, or some combination of automatic and manual selection, for example. In a case that queries are selected automatically, any criteria can be used. For example, queries can be selected based on popularity of a query, or lack thereof, such as might be determined based on frequency of use of a query. To illustrate by way of non-limiting example, a set of queries having differing popularity can be selected,

such as by selecting a percentage of queries having a high degree of popularity, a percentage of queries having a medium level of popularity, and a percentage of queries having a low degree of popularity. In addition, guidelines can be established for obtaining judgment input from human judges concerning search results (e.g., a number and type of questions to be asked of the judges, the types of responses to each type of question, and the significance of the responses), confidence intervals to be used with the results, optimum combinations of settings of variable, and how the variables affect relevancy and coverage in the optimum region, and what variables are most significant in the optimum region.

[0029] In accordance with at least one embodiment, a confidence interval (CI) is an estimated range of values which is likely to include an unknown population parameter. The higher the CI, the more samples (e.g. queries) are needed. A variable (which is also referred to herein as a “knob”) represents a parameter of the search engine which can be used to alter the processing (e.g., retrieval, sorting, result ranking, etc.) performed by the search engine. Examples of parameters, or knobs, include, without limitation, parameters used to weight the information retrieved by a search engine. For example, a title parameter can comprise a predefined value used to weight a search result (e.g., a document) having a corresponding title which contains one or more query terms higher, or lower, than other results having corresponding titles which do not contain a query term. Other examples of parameters include, without limitation, date, popularity, proximity, and literal weighting. A date parameter can be used to boost (e.g., add to) a score or ranking associated with newer documents. A popularity parameter can be used to boost a score, or ranking, of “popular” documents, e.g., a popular document can be determined based on a number of selections or views of the document. A proximity parameter can be used to boost a score, or ranking, associated with a document whose contents include query terms in close proximity to one another. A literal parameter can be used to boost a score, or ranking, of a document which exactly matches a query term. The value of one or more parameters, or knobs, can be changed to “tune” a search engine, so as to find an optimal configuration of the search engine, e.g., so as to identify a configuration in which search engine performance (e.g., relevance of search results generated by the search engine) is the highest. Embodiments of the present disclosure use various metrics (e.g., DCG) described herein to determine search engine performance.

[0030] Embodiments of the present disclosure can use parameters in addition to, or as a replacement for, those described above. Examples of other parameters that can be used in accordance with one or more embodiments of the present disclosure include, without limitation, parameters to identify information associated with a document in which to search (e.g., title, anchor text, document body, etc.), parameters which can be used to exclude/include documents based on features of the document (e.g., document layout, appearance, and/or features indicative of spam). In addition, embodiments of the present disclosure can use parameters to use different search indices, or index sizes.

[0031] Embodiments of the present disclosure can be used by search engine developers or engineers who develop and/or adjust (or tunes) a search engine, for example. There are many areas of a search engine that can be adjusted, or tuned, which can impact performance, in either a positive or negative way. For example, and as described above, engineers can boost document ranking based on a popularity measure or date associated with a document. Boosts, or weightings, can impact relevance of search results and/or rank, order and/or presentation of search result items, for example. The impact might be a positive or it could be negative. Embodiments of the present disclosure can be used to experiment with design alternatives and then to evaluate the search engine's performance in order to identify an optimal configuration. For example and using one or more embodiments, an engineer, or more generally a user, can study and effect, or effects, of individual and/or combined adjustments to a search engine, and/or interaction(s) that may exist between one or more such adjustments. In addition to different configurations of the same search engine, embodiments of the present disclosure can be used to evaluate performance using more than one search engine. Embodiments of the present disclosure can be used to compare multiple configurations of the same search engine, or multiple configurations of different search engines, for example. It should be apparent that embodiments disclosed herein can be used for a pairwise comparison of two information retrieval engines (e.g., search engines), however, the embodiments are not limited to such a pairwise comparison. Rather, embodiments of the present disclosure can be used to evaluate any number of engines (e.g., different engines or different configurations of one or more engines) using the same or different settings of one or more "knobs". As is discussed herein, various metrics can be used to perform one or more evaluations, even in a case that a complete set of judgment input is unavailable. In a case that incomplete set of judgment input is available (e.g., a cache hit is

less than 100%), at least one metric discussed herein can be used to predict relative performance of one or more engines under evaluation.

[0032] Step 102 comprises defining QBs and the set of queries to be included in a QB in accordance with system design and setup decisions made in step 101. Step 102 results in one or more sets of queries, each set corresponding to a QB, and each QB comprising one or more queries. In addition, a sampling period can be established to identify the interval between generation of search results using a given QB. The sampling period can be used to determine when to run a QB to obtain search results. A QB can also include an identification of a type of judgment input to be obtained from a judge, or judges.

[0033] The queries specified for a QB are run using one or more search engines to generate query results, at step 103. The queries specified for a QB can be run by one or more different search engines, and/or different configurations of the same search engine, to generate the output corresponding to each of the search engines, or input a QB to different configurations of the same search engine to generate output for each of the different configuration of the search engine, for example.

[0034] In accordance with one or more embodiments of the present disclosure, query results generated by a search engine are analyzed to identify possible duplicate results. In some cases, results can have an identifier. In some cases, the identifier uniquely identifies a result and can provide a mechanism to identify duplicates. In some cases, a document identifier may not necessarily uniquely identify a document. For example, in a case that a result corresponds to a document, e.g., a web page, retrieved from a web site, a universal resource locator (URL) can be used to identify the document. While the same document might be available using multiple different URLs, a document's URL can be used to identify duplicates based on the same or a similar URL, for example. As an alternative, or in addition, to use of an identifier, the contents of a result (e.g., a web page returned by a web search engine) can be examined, and/or metadata associated with result can be examined. In a case that result contents are examined, some portion of the result contents might be ignored (e.g., ad content, certain hypertext tags, etc.).

[0035] In addition to storing the query results corresponding to a given QB, information such as the search engine used, the data source/provider searched, temporal information (e.g., date and time of the search), and the number of results generated can be stored for a given

QB. Coverage analysis can be performed on the query results generated for a given query. For example, coverage analysis can involve a determination as to the number of results generated for each query for a single run, or aggregated across multiple runs, of the QB, and/or the number of times, and/or frequency with which, the QB is used to generate query results.

[0036] Step 105 comprises obtaining, from one or more judges, judgment input corresponding to query results generated from a given QB. Step 104 comprises defining a performance test using a QB. The performance test involves identifying a QB and two “crawls” corresponding to the QB. Each crawl identifies a search engine and a data source/provider. In addition, each crawl has an associated result set corresponding to the set of queries associated with the specified QB. The result set associated with each crawl and judgment input associated with result items in the result sets can be used to evaluate a search engine’s performance. For example and using a QB and received judgment input corresponding to the result sets, the specified crawls can be compared based on one or more performance indicators, and one or more metrics associated with the one or more performance indicators, search engine performance can be evaluated, at step 106.

[0037] To illustrate by way of a non-limiting example, it is assumed that a performance test is defined, at step 104, which specifies a QB, first crawl associated with a first search engine instance, and a second crawl associated with a second search engine instance. The search engine instances can correspond to different search engines, or can be different configurations of the same search engine, for example. In the example, it is assumed that each crawl has an associated set of results, also referred to as a result set, which comprises one or more result items corresponding to the set of queries, or query set, specified for the QB. In addition, it is assumed that there is some judgment input associated with result items in the result set for both of the crawls. The judgment input was received from at least one judge at step 105.

[0038] At step 106, the received judgment input is used to analyze and evaluate performance based on the first and second crawls. To further illustrate using this example, for a given query in the first crawl, there exists judgment input which identifies relevance of a given query result item to a query. The same query is used in the second crawl to generate a subsequent set of search results, which results has one or more search result items in common

with the first crawl's set of search results. Since the same query was used, the same relevance judgment input can be used at least with respect to the one or more search result items that appear in both crawls, to determine a relevance of the second crawl's search results to the query.

[0039] In addition and in accordance with one or more embodiments, the relevance judgment input corresponding to the first crawl's search results can be used to estimate relevance data for the second crawl's search result items that are absent from the first crawl's search results, and/or to determine an overall relevance score for the second crawl, in step 106. While relevance is used in this example as an indicator of performance, it should be apparent that other indicators can be used to gauge performance, such as distance and coverage indicators, which are discussed herein. In addition, and is discussed herein, it should be apparent that one or more metrics can be determined for a given indicator.

[0040] Figure 2 provides an example of an architecture overview comprising components for use in performance evaluation in accordance with one or more embodiments of the present disclosure. Although not shown, a system design component can be used in accordance with one or more embodiments, which can allow for identification of a methodology, and general guidelines, which methodology and guidelines can be used to configure one or more other components. For example, QB creation 201 can be configured using system design information which identifies the number of queries used in a QB, as discussed herein at least with reference to Figure 1.

[0041] QB creation 201 is configured to define one or more QBs, each of which has an identified query set comprising one or more queries. Crawl and coverage analyzer 203 is configured to request crawls to be performed by search engine(s) 204 of search provider(s)/data source(s) 205, based on the QBs defined by QB creation 201, system configuration information and/or previous crawls identified, for example, using query log(s) 202.

[0042] Crawl and coverage analyzer 203 is configured, in accordance with one or more embodiments, to interface with one or more instances of search engine 204, such that a query is forwarded to search engine 204, which queries one or more data providers/sources 205 to retrieve query results and the query results are returned. Crawl and coverage analyzer 203 uses query log(s) 202 to analyze coverage, and to determine whether or not to request one or

more subsequent crawls, e.g., based on a sampling period. Coverage analysis performed by crawl and coverage analyzer 203 can include coverage with respect to time, e.g., frequency of occurrence of a crawl, and/or with respect to a number of search results, and/or hits, in a result set, for example.

[0043] Judgment system 207 is configured to provide query set results to one or more of judges 208 to obtain judgment input, which input can be stored in judgment database 206. Figure 3 provides an example of a portion of a judgment input user interface for use in accordance with one or more embodiments of the present disclosure. A query 301 is displayed, along with one or more result items 302. Each instance of result item 302 comprises a hyperlink 312, which allows the judge to view a landing page associated with the result item 302, and a text portion 322 of the search result item 302. Although one result item 302 is shown, it should be apparent that there can be multiples instances of result item 302 corresponding to query 301.

[0044] In accordance with one or more embodiments, judgment system 207 can assess reliability of judgment input. Some queries and/or results can be ambiguous which may result in inconsistent judgment input from multiple human judges. Judgment system 207 can be used to identify these queries and results, and remedial measures can be taken including removal of a query and/or result from a QB.

[0045] Intra-judge agreement is another reliability measure that can be assessed using judgment system 207 in accordance with embodiments of the present disclosure. Judgment 207 can be used, for example, to determine whether or not a judge 208 is giving different relevance judgment input to the same result at different times. Inconsistencies in judgment input provided by a judge 208 can raise a question as to the reliability of the judgment input. In addition, a judge 208 may have a bias that can impact the judgment input for that judge. Such an inconsistency can be detected based on an examination of judgment input across judges and/or examination of inconsistencies in judgments received from a single judge, for example. Judgment system 207 uses statistical techniques to systematically detect inconsistent judges and/or results. In accordance with at least one embodiment, judgment system 207 uses one or more of percent agreement (PA), correlation measure (CM), and many-facets Rasch model (MFRM) testing to detect inconsistent judges and/or inconsistent judgment input. PA and MFRM can be used, in one or more embodiments, to identify judges

who are making inconsistent judgments and/or results that received inconsistent judgments. CM can be used, in one or more embodiments, to identify inconsistent judges.

[0046] In accordance with one or more embodiments, one or more tests can be used to evaluate inter-judge agreement, or consensus among human judges, and/or intra-judge, or consistency of an individual judge. The PA test, or statistic, can be used to determine a percentage of identical judgments among the judges providing input for a given result. The CM test can be used to determine an average correlation of judge's judgments relative to other judges. The MFRM test can be used to estimate and evaluate differences in judge severity and screen judges whose judgments lack variation or self-consistency. The PA and MFRM tests can be used to identify judges who are making inconsistent judgments and results that have received inconsistent judgment input, for example. CM can be used to identify inconsistent judges.

[0047] In accordance with one or more embodiments, a judgment set is constructed for a random set of queries, "Q". The judgment set is provided by J judges, each judge, j, evaluating a subset of queries in Q, Q_j . In accordance with at least one of the embodiments, the query subset, Q_j , for each judge, j, is done such that each query, q, has R results and is judged by at least D judges. That is, for a query, q, of a query set Q_j , each judge, j, enters R judgments. The judgment, or judgment input, can comprise a value, x, taken from a scale, C, which comprises categories representing values of x.. For example, in a case that the judgment input represents relevance using a tertiary (or three-category or value) scale, C, which comprises value categories of "2", "1" and "0", a result can be rated as highly relevant (e.g., c-value of "2"), relevant (e.g., c-value of "1"), or nonrelevant (e.g., c-value of "0"). Thus, in the example described, there can be a judgment set consisting of tuples $\langle q, r, j, x \rangle$ where x is the judgment input value input by judge j for a result r and a query q. The total number of judgments can be determined to be: total judgment input = total number of queries, Q, multiplied by the total number of results, R, multiplied by the total number of judges, D, judging the results.

[0048] Embodiments of the present disclosure identify a filter, F_q , that returns a set of queries in Q that are determined to be too subjective or difficult to evaluate by the judges, and a filter, F_j , that returns a set of judges in J determined to provide judgment input inconsistent (e.g., judgment input is too lenient, conservative, and/or unreliable) with other judges in J.

[0049] In accordance with one or more embodiments, a PA statistic counts raw matching scores for each query result pair, $\langle q, r \rangle$, CM determines an average correlation of a judge's judgments with those of other judges, and MFRM compares a severity of all judges on all items, even if they did not rate the same items. The PA can be used to evaluate a degree of agreement of judgments among judges for a particular result. Let k be the total number of categories ($1 \leq k \leq C$). For each $\langle q, r \rangle$ pair, or case, m , let n_{km} be the number of times category k is applied. For example, if $\langle q, r \rangle$ is rated 5 times and received ratings of 1, 1, 1, 2, 2, then $n_{1m} = 3$ and $n_{2m} = 2$. Let n_m be the total number of ratings made on case m , for the $\langle q, r \rangle$ pair:

$$n_m = \sum_{k=1}^C n_{km} \quad (1)$$

[0050] For each case m , or for each $\langle q, r \rangle$ pair, the number of agreements on rating level k is $n_{km} \times (n_{km} - 1)$. Using the above ratings example, the agreement on rating $k=1$ is $3 \times 2 = 6$, and the agreement on rating $k=2$ is $2 \times 1 = 2$. The number of agreements across all categories is:

$$SP = \sum_{k=1}^C n_{km} \times (n_{km} - 1) \quad (2)$$

[0051] The total possible number of agreement on case $\langle q, r \rangle$ is:

$$SAP = M \times (M - 1), \quad (3)$$

[0052] where $1 \leq M \leq J$. The percent agreement for case m , for a given $\langle q, r \rangle$ pair, is defined as:

$$PA_m = \frac{SP}{SAP}, \quad (4)$$

[0053] where $1 \leq m \leq Q \times R$. Using the same example as above, the percent agreement is:

$$\frac{3 \times 2 + 2 \times 1}{5 \times 4} = 0.4 \quad (5)$$

[0054] The filter, F_q , which can be used to filter queries that received inconsistent judgments, can be defined as:

$$F_q = \begin{cases} 1 \dots \text{if } PA_m \leq 0.5; \\ 0 \dots \text{Otherwise} \end{cases} \quad (6)$$

[0055] To filter judges based on determined inconsistent judgments, a set of $\langle q, r \rangle$ for which the percent agreement is high ($PA_m \geq 0.5$) is determined. This set of cases can be and the corresponding judgments can be referred to as “golden” sets. A percentage of matches can be computed between each judge’s judgments for the $\langle q, r \rangle$ and the golden sets. Let M be the total number of cases in the golden set, and A be the number of matches, a judge’s agreement statistic with the “golden” set can be defined as:

$$JA_j = \frac{A}{M}, \quad (7)$$

[0056] where $1 \leq j \leq J$. The filter F_j for inconsistent judges can be defined as:

$$F_j = \begin{cases} 1 \dots \text{if } JA_j \leq 0.5; \\ 0 \dots \text{Otherwise} \end{cases} \quad (8)$$

[0057] In accordance with one or more embodiments, a correlation measure comprises an average correlation of a judge’s judgments with those of every other judge. High values of correlation between two judges indicate a similarity of their judgments. Judges who are known to be reliable can be identified as “golden judges”. High correlation values between a judge and a golden judge can be used to identify a reliable judge and results.

[0058] The CM uses correlations among judges to identify inconsistency. In accordance with at least one embodiment, each judge judges every case. In such a case, the total number of judgments is $Q \times R \times J$. The CM correlates two judges, x and y , at a time, with $Q \times R$ judgments from each judge, the judgments being referred to as x_i and y_i , where $1 \leq i \leq Q \times R$. In accordance with one or more embodiments, a standard Pearson’s correlation can be used to determine a correlation between the two judges using the following, for example:

$$r_{xy} = \frac{\sum_{i=1}^{Q \times R} (x_i - \bar{x}) \times (y_i - \bar{y})}{(n-1) \times s_x \times s_y}, \quad (9)$$

[0059] where s_x and s_y are the standard deviation of x and y . The total number of inter-judge correlation can be determine as follows:

$$N = \frac{J \times (J-1)}{2} \quad (10)$$

[0060] A sum of all of the inter-judge correlation can be determined as follows:

$$r_{total} = \sum_{i=1}^J \sum_{j=i}^J r_{ij} \quad (11)$$

[0061] An average inter-judge correlation can be determined as follows:

$$\bar{r} = \frac{r_{total}}{N} \quad (12)$$

[0062] A variance of inter-judge correlation can be determined as follows:

$$\text{var}(r) = \frac{\sum_{i=1}^N (r_i - \bar{r})^2}{N} \quad (13)$$

[0063] A standard deviation can be determined as follows:

$$\sigma_r = \sqrt{\text{var}(r)} \quad (14)$$

[0064] An average correlation for judge j can be determined as follows:

$$r_j = \frac{\sum_{i=1}^J r_{ij} - 1}{J} \quad (15)$$

[0065] A filter for inconsistent judges can be defined as follows:

$$F_j = \begin{cases} 1 \dots \text{if } r_j < \bar{r} - 2 \times \sigma_r, \text{ or } r_j > \bar{r} + 2 \times \sigma_r ; \\ 0 \dots \text{Otherwise} \end{cases} \quad (16)$$

[0066] In accordance with one or more embodiments, a Many Facets Rasch Model (MFRM) can be used to estimate differences in judge severity and screen judges whose judgments lack variation or self-consistency. The MFRM can be used to provide estimates of the consistency of observed rating patterns. Fit can be used as a quantitative measure of the discrepancy between the statistical model and the observed data. An expectation is that highly relevant results achieve consistently higher scores than less relevant results, and a residual difference between expected and observed scores is the basis of fit analysis. Two fit statistics can be reported: an infit score and an outfit score.

[0067] The MFRM test can use one or more “facets” in evaluating a judge and/or judgment input. Examples of facets that can be used in one or more embodiments include, without limitation, inherent relevancy of a search result for a given query, difficulty of the questions asked, severity of a judge, and difficulty of rating scale. The MFRM is based upon the assumption that one should use all of the information available from all judges (including discrepant ratings) when attempting to create a summary score for each search result being judged. It is not necessary for two judges to come to a consensus on how to apply a scoring rubric because differences in judge severity can be estimated and accounted for in the creation of each result’s final score. When a search result is rated by a judge, the log odds (logit) of it being rated in category x is modeled by the following:

$$\log\left(\frac{P_{nj k}}{P_{nj(k-1)}}\right) = \theta_n - \beta_j - \xi_k, \quad (17)$$

[0068] where $P_{nj k}$ is a probability of search result n being awarded a rating of k when rated by judge j , $P_{nj(k-1)}$ is a probability of result n being awarded a rating of $k-1$ when rated by judge j , θ_n is a relevancy of search result n , β_j is a severity of judge j , and ξ_k is a difficulty of achieving a score within a particular score category k .

[0069] In accordance with one or more embodiments of the present disclosure, a model such as that set forth in expression (17) is used to configure a Rasch measurement software utility, such as that provided by Winsteps®, which operates on the judgment input using the model to generate measures such as those shown in Figure 7.

[0070] The example output shown in Figure 7 reflects from raw input received from thirty-four judges, each of whom provided relevancy input for the top five results of thirty queries generated from a search engine. The results were randomly mixed and shown to the judges. Each judge was asked to provide judgment input as to the relevancy of each search result for each query based on a five point scale: 1) excellent, 2) good, 3) fair, 4) bad, and 5) no judgment. The results were recorded and were input to the Rasch software utility, which operated on the raw input using the above-identified model.

[0071] The measure column 704 lists normalized logit values for severity of judges. The maximum value in column 704 identifies the most lenient judge. The minimum value in column 704 identifies the most severe judge. In this case, row 702 corresponds to the judge

(i.e., judge number 20) who gave the most lenient scores, and row 706 corresponds to the judge (i.e., judge number “12”) who gave the most severe score.

[0072] The inventors have determined that a judge’s severity measure is highly correlated with the difference of the judge’s average score and the overall average score. The “Infit MnSq” column 708 provides a consistency measure for each of the judges. For all the judges, the mean value of “Infit” is 1.01 with a standard deviation of 0.44. As one example of a guideline, if the “Infit” value goes outside of the value of $\text{Mean} \pm 2 \times \text{Std.Dev.}$, the judge is considered to not be applying the scoring criteria consistently. In the example shown in Figure 7, the accepted range of “Infit” values is $1.01 \pm 2 \times 0.44$, or [0.13, 1.89]. Using the example guideline, row 707 (i.e., judge number 27) is identified as a “misfit”. The inventors determined that this judge’s scores for each result significantly deviated from the average score of each result determined across all of the judges, sometimes by as much as 127%.

[0073] In accordance with one or more embodiments, an Infit mean square is an information-weighted chi-square statistic divided by its modeled degrees of freedom. An example of an equation for determining an Infit mean square for a given judge, j , is as follows:

$$V_j = \frac{\sum_{n=1}^N \sum_{j=1}^J (x_{nj} - E(x_{nj}))^2}{\sum_{n=1}^N \sum_{j=1}^J \text{var}(x_{nj})}, \quad (18)$$

[0074] where $\text{var}(x_{nj})$ can be determined as follows:

$$\text{var}(x_{nj}) = \sum_0^m x - E(x_{nj}))p(x_{nj}), \quad (19)$$

[0075] where, under the Rasch model conditions, $\text{var}(x_{nj})$, which comprises a modeled variance of an observation around its expectation. In equation (19), $\text{var}(x_{nj})$ is the variance of the observation awarded to result n by judge j , m is the highest numbered category for observations, $p(x_{nj})$ is the expected value of the observation, $E(x_{nj})$ is the probability that result n will be observed by judge j in category x .

[0076] An Infit scoring can be used to compare the sum of squared ratings residuals with their expectation. In accordance with one or more embodiments, a range of the Outfit and In-

fit scoring is 0 to infinity, with a modeled expectation of 1.00 and a variance inversely proportional to the number of independent replications in the statistic referenced. The Infit scoring is sensitive to an accumulation of on-target deviations that are less or more consistent than expected. The Outfit scoring is sensitive to off-target responses due to carelessness or misunderstanding. In one or more embodiments, the Infit scoring is used rather than the outfit scoring. In accordance with disclosed embodiments, an outfit mean-square can be a chi-square statistic divided by its degrees of freedom, as follows:

$$U_j = \sum_{n=1}^N \sum_{j=1}^J \frac{(x_{nj} - E(x_{nj}))^2 / \text{var}(x_{nj})}{NJ}, \quad (20)$$

[0077] where N is the number of search results and J is the number of judges.

[0078] Referring again to Figure 3, a judgment input 303 portion includes one or more questions and associated response options. In the example shown in Figure 3, judgment input 303 comprises “per item” judgment input, whereby a judge is asked to provide judgment input for each result item in a result set for a given query. For example, question 305 prompts a judge to provide a response, in response portion 306, concerning relevance of result item 302 to query 301. The judge has an option of accepting the result 302 as relevant, conditionally relevant, rejected as not relevant, rejected for another reason, or no judgment offered. An instance of pull-down menu 310 allows a judge to make a selection to further refine a response. The contents of a pull-down menu 310 can be specific to a given question and response, for example.

[0079] In the example shown in Figure 3, the judge is asked to also indicate a relevance of the landing page to the query in question 307. The judge can view the landing page by selecting hyperlink 312. As with question 305, the judge has an option of accepting the result 302 as relevant, conditionally relevant, rejected as not relevant, rejected for another reason, or no judgment offered, in response portion 308. Pull-down menu(s) 310 allows a judge to further refine a response. As indicated above, the contents of a pull-down menu 310 can be specific to a given question and response.

[0080] In accordance with one or more embodiments, result items can be scrambled and displayed in a generic way, so as to reduce a possibility of bias toward a particular search engine, for example. Examples of other types of judgment input include, without limitation,

“per set” and “side-by-side”. With regard to “per set” judgment input, a judge provides judgment input on a result set basis. For example, a judge can be asked to provide judgment input for a result set based on a first page of results. A judge can take into account features such ranking of results in the result set and/or provider/source diversity associated with a result set, etc. With regard to side-by-side judgment input, a judge is requested to review two sets of search results side-by-side, e.g., the top 10 results), and provide judgment input as to which side the judge prefer. The judge can also be asked to provide an explanation/reason for the preference. To minimize bias, the result sets can be presented in a manner so as to minimize a judge’s ability to identify a search engine by its output.

[0081] Referring again to Figure 2, judgment system 207 can store judgment input in judgment database 206, and/or aggregate judgment input and forward the aggregated judgment input to analysis database 210. Judgment input can be aggregated across performance tests, and can include recent score, average score, maximum score and minimum score, for example. Multiple judgment input for a query result can be due to judgment input from more than one judge, or the same result item being judged across various performance tests, for example. In addition, judgment system 207 can forward non-aggregated judgment input to analysis database 210. In fact and although the databases are shown separately in Figure 2, it should be apparent that analysis database and judgment database 206 can comprise the same database.

[0082] Analysis Database 210 comprises data used by data analysis system 209 to generate one or more metrics corresponding to indicators output via performance test definition and evaluation system 211 to a user. Performance test definition and evaluation system 211 provides a user interface that allows a user to select an existing QB, first crawl and second crawl, or to request/define a new QB, a new first crawl and/or a new second crawl, so as to define a performance test. In accordance with one or more embodiments, the QB selection filters the crawl selection available to the user, to crawls associated with the QB. A QB comprises a set of queries, each of which can be entered by the user or selected from query log(s) 202. Each QB has a name, and can have other information such as creation date, etc. In addition, the first and second crawl specified for a performance test comprises a result set for each query in a query set corresponding to the selected QB.

[0083] In addition and in defining a performance test, a user can identify one or more indicators, and one or more metrics for a given indicator, of performance for use with the performance test. Examples of indicators/categories of metrics include distance, relevance and coverage.

[0084] Distance metrics provide an overview of the degree to which result sets associated with the two crawls vary. For example and with regard to a specific query and the result sets for the query in the two crawls, a metric can provide a measure of a degree to which the ordering of the results vary in the two result sets, or measure of a degree to which the two results contain different result items, etc. Examples of distance metrics include, without limitation, Spearman footrule, Kendall Tau, and set overlap metrics.

[0085] A Spearman Footrule metric comprises a sum, across n result items in a first result set $R1$, of the absolute difference between the rank of the i th result item in result set $R1$ and the rank of the same result item in a second result set, $R2$. A normalized footrule distance can be determined by dividing the sum by a normalization value. An example of a normalization value which can be used in one or more embodiments is a square of a maximum shift value, S , divided by two, or $(|S| * |S|) / 2$. A normalized footrule distance can range from zero and one, where a zero value indicates that, in the two results sets, the result items are ranked the same, and a value of one indicates that the results are either ranked in reverse order or the results in the two result sets are different. The normalized footrule distance can be used to identify a difference between the result sets $R1$ and $R2$ that might lead to further examination of one or more queries and corresponding results sets in one or both of the crawls, for example. Assuming, for example, that result set $R1$ was provided using a search engine 204 instance prior to a change and that result set $R2$ was provided using a search engine 204 after a change to the search engine instance. A high normalized footrule distance value can indicate that the change had a significant impact on performance of the search engine 204 instance, and further examination is warranted to determine whether or not the change in performance is preferred.

[0086] Figure 4A provides a Spearman footrule distance example in accordance with one or more embodiments of the present disclosure. In the example shown, $R1$ corresponds to list 401 and $R2$ corresponds to list 402. With regard to the first item, A , in list 401, it can be seen that item, A is shifted from first position in list 401 to third position in list 402. It can be said

that the absolute difference, or shift, of item *A* as between lists 401 and 402 is 2. In addition, the differences, or shifts, relative to items *B*, *C* and *D* can be said to be 1, 1 and 2. The sum of the distances is 6, i.e., $2+1+1+2$. A normalization value can be $(4*4)/2$, or 8, since a maximum shift value, *S*, can be four. A normalized value using the normalization value of 8 is $6/8$, or 0.75.

[0087] In accordance with one or more embodiments, a Kendall Tau metric used in connection with a distance indicator comprises a count of the number of pair-wise disagreements between two lists, *R1* and *R2*, which can be expressed as follows:

$$K(R1, R2) = |\{(i, j) | i < j, R1(i) < R1(j), \text{ but } R2(j) > R2(i)\}|, \quad (21)$$

[0088] where *i* and *j* are positions in *R1* and *R2*. A non-zero distance for a given pair of items is indicated where a position of one result item, *i*, in *R1*, or *R1(i)*, is less than a position of another result item, *j*, in *R1*, or *R1(j)*, but in *R2*, the position result item, *i*, or *R2(i)* is greater than the position of result item, *j*, in *R2*.

[0089] Figure 4B provides a Kendall Tau distance determination example in accordance with one or more embodiments of the present disclosure. In the example, which uses the same lists as in Figure 4A, *R1* corresponds to list 401 and *R2* corresponds to list 402. With regard to the first item, *A*, from list 401, it can be seen that the position of item *A*, i.e., in the first position, is less than the position of item *B*, i.e., in the second position, in list *R1*. In list 402, however, *A*'s position, i.e., in the third position, is greater than *B*'s position, in the first position. The *AB* pair is considered to be a pair whose items disagree in lists 401 and 402. As indicated in Figure 4B, there are two other pairs, i.e., *AD* and *CD*, that disagree. The Kendall Tau distance, or $K(R1, R2)$ for the example shown in Figure 4B is therefore 3. A normalization Kendall Tau distance can be determined by dividing $K(R1, R2)$ by a maximum possible value, e.g., $(|S| / 2)$, e.g., $6/2$, or 3.

[0090] A further discussion of Spearman footrule and Kendall Tau distance metrics can be found in an article entitled, *Rank Aggregation Methods For The Web*, by Cynthia Dwork, Ravi Kumar, Moni Naor, and D. Sivakumar, *In proceeding of the Tenth International World Wide Web Conference, 2001*, and in an article entitled *Comparing Top k lists*, by R. Fagin, R. Kumar, and D. Sivakumar, *In proceedings of the ACM-SIAM Symposium on Discrete Algorithms, 2003*, both of which are incorporated herein by reference.

[0091] Set overlap is the number of overlapping pairs between two result sets. Assuming a set of two lists, set overlap compares a pair of items identified as the items at a given level (e.g., row) in the two lists. The result is the number of item pairs taken from the two lists that are the same. For example, if the first set is $\{x, \text{with } N \text{ or more elements}\}$ and the second set is $\{y(i), \text{with } N \text{ or more elements}\}$, overlap is defined as the number of common elements in $\{x(i), 1 \leq i \leq N\}$ and $\{y(i), 1 \leq i \leq N\}$ (i.e., where $x(i)$ and $y(i)$ match) divided by N . The comparison of the lists can involve all of the items in the lists, or some number up to a certain level. For example, in a case that N equals 100, set overlap can be performed on all 100 items, or the first M items.

[0092] In accordance with one or more embodiments of the present disclosure, a relevance indicator comprises a measure of a precision corresponding to one or more result sets. Precision can correspond to a determination of a number of the results in a result set are relevant to a given query, or topic, for example. Recall can correspond to a determination ratio of a number of relevant results returned to the number of results returned for a given query, or topic, for example.

[0093] A precision metric can be computed for a given query and can comprise a mean average precision metric, and a discounted cumulative gain, for example. A precision metric, referred to as a rank-based precision metric, can be determined based on result item rank. For example, result items in a result set can be ranked from 1 to n (where n can be the total number of result items, or some number less than the total number of result items). The ranking can be based on a relevance score determined using judgment input, for example. A result item can be considered to be the most relevant to a query, and the next result item is the second-most relevant to a query, etc. A parameter associated with a rank-based precision metric can identify a threshold position (or rank) from 1 to n , which represents a position in the ranking of the result items, such that result items at and above the threshold are used in determining the rank-based precision metric, for example. For example, in a case that n is equal to 20 and the threshold is 5, result items 1 to 5 are used to determine a precision metric value, and result items 6 to 20 are excluded from the determination.

[0094] In accordance with one or more embodiments of the present disclosure, a precision metric can be an aggregate. For example, query-level (rank-based or otherwise) precision metrics corresponding to queries in a given crawl can be aggregated to yield a crawl-level

(rank-based or otherwise) precision metric. As with a query-level rank-based precision metric, a rank-based precision metric corresponding to a given crawl can represent a relevance score for result items up to a threshold position.

[0095] Another example of an aggregate precision metric at the crawl-level is an average of relevance scores for all result sets corresponding to the queries in a crawl's query set. In a case that a relevance score for a given result item is indicated as a 0, if irrelevant, and 1, if relevant, an averaged precision metric can be the average of relevance scores (rank-based or otherwise) in a result set, such averaged precision metric can range from 0 to 1, where 0 indicates that there are no relevant results, and 1 indicates that all of the result items are relevant. In accordance with one or more embodiments, an averaged precision metric is also referred to as a precision level, which precision level can be determined for each of the crawls in a performance test.

[0096] In accordance with one or more embodiments, a discounted cumulative gain (DCG) precision metric can be used as a relevance indicator corresponding to the first and second crawls. In determining a DCG precision metric value, precision (e.g., a relevance score) at each position is used as a "gain" value measure for its ranked position in the result and is summed progressively from position 1 to n . A discounting function progressively reduces the document score as the rank keeps increasing but not steeply (e.g., division by the log of the rank). The cumulated gain vector is defined recursively as

$$DCG[i] = \begin{cases} G[1], & \text{if } i = 1; \text{ else} \\ DCG[i-1] + G[i]/\log(i+1), & \text{if } i > 1 \end{cases} \quad (22)$$

[0097] where i is a rank position and $G[i]$ is a scale value at position i .

[0098] An idealized DCG value can be computed by reordering the results, with the relevant results ranked above the irrelevant results, and an idealized DCG (IDCG) value can be computed as above. In addition, a normalization of an idealized DCG can be determined. For example, let $\langle v_1, v_2, \dots, v_k \rangle$ represent the DCG vector and $\langle i_1, i_2, \dots, i_k \rangle$ represent the idealized DCG vector, then the normalized DCG is given by $\langle v_1/i_1, v_2/i_2, \dots, v_k/i_k \rangle$. By averaging over all of the result sets for a given crawl, a DCG precision metric can be determined for the crawl, such that the performance of one crawl can be compared to another. The average vectors can be visualized as gain by rank graphs. A further discussion of

cumulative gain can be found in an article entitled *Cumulated Gain-Based Indicators Of IR Performance*, by Kalervo Jarvelin, Jaana Kekalaninen, *In proceedings of the ACM Transactions on Information systems, 2002*, the contents of which are incorporated herein by reference.

[0099] In accordance with one or more disclosed embodiments, the DCG is used in a case that there is no prior knowledge of relevance in connection with a given one or more search engines. Alternatively and in a case that there is prior knowledge that relevance of the search engines is relatively good, the LinearFT metric can be used as it is likely to be more robust than the DCG metric in terms of preserving system ranking and accuracy based on significance tests under incomplete judgments

[00100] The DCG preserves system ranking and has the best accuracy based on significance tests under various levels of incomplete judgments. A system ranking can be defined as an ordered list of search engines sorted by decreasing evaluation measure score. Significant tests are based on hypothesis testing. For a pair of search engines, the null hypothesis (H0) is that the relevance metrics are equal for both engines. If confidence in the hypothesis (reported as a p-value) is ≤ 0.05 (at 95% confidence level), the null hypothesis is typically rejected, i.e., the two engines have statistically significant difference based on the relevance metric. To test the hypothesis, the value of the DCG metric for each query is calculated. For each search engine, there is a vector of metric values for all the queries. The two vectors of metric values from the two engines can be compared using a pairwise Wilcoxon test to determine the p-value.

[00101] The Discounted Cumulative Gains (DCG) metric is defined as described above. For example, if the scale value for a relevant document is 1, and the scale value for a nonrelevant document is 0, the DCG metric is the following:

$$\text{Binary DCG}[i] = \sum_r \frac{1}{\log(i+1)}, \quad (23)$$

[00102] where r is a relevant document, and i is the rank position of the relevant document. In essence, it is the summation of the inverse log rank for all relevant documents. The binary scale DCG metric is more robust under incomplete judgments compared to the DCG metric with more than two scale values.

[00103] The binary DGC uses a binary relevance scale, by which a document, x , has a designated relevance score of 0 or 1, e.g., a document, x , has a relevance scoring of $\text{rel}(x)=1$ if it is relevant, and a scoring of $\text{rel}(x)=0$ if it is not relevant. An example of a tertiary scale which uses a three category scoring scale, by which in a case that document, x , is highly relevant its corresponding scoring is $\text{rel}(x)=2$, $\text{rel}(x)=1$ in a case that it is relevant, and $\text{rel}(x)=0$ in a case that it is not relevant. The inventors determined that there is no significant difference between the binary and tertiary DCG values.

[00104] In accordance with at least one disclosed embodiment, binary judgments assume $\text{rel}(x) \in \{0, 1\}$ such that $\text{rel}(x) = 0$ denotes x is not relevant to the topic and $\text{rel}(x) = 1$ denotes x is relevant to the topic. In a case of a scaled judgment (e.g., a five point scale: 1) excellent, 2) good, 3) fair, 4) bad, and 5) no judgment), the value of $\text{rel}(x) > 0$ to indicate the degree of relevance. In a case of incomplete judgments, there is at least one x in which the $\text{rel}(x)$ is unknown, e.g., x has not been judged for a given query, or topic.

[00105] If there is prior knowledge that relevance of the search engines is relatively good, the LinearFT metric is likely to be more robust than the DCG metric in terms of preserving system ranking and accuracy based on significance tests under incomplete judgments. The LinearFT metric can be determined as follows:

$$\text{LinearFT} = \sum_r \frac{1}{\log(i+1)} - \sum_{nr} \frac{1}{\log(i+1)}, \quad (24)$$

[00106] where r is a relevant document, nr is a nonrelevant document, and i is the rank position of the document. In essence, it is the difference between the summation of inverse log rank for all relevant documents and all nonrelevant documents.

[00107] In the above example, the difference between the DCG metric and the LinearFT metric is the treatment of nonrelevant documents. In the DCG metric, the nonrelevant documents are ignored. However, in the LinearFT metric, there is a penalty for known nonrelevant documents. When relevance of the search engines is already good, the penalty for known nonrelevant documents provides additional differentiating power. On the other hand, when relevance is low for the search engines, the LinearFT metric may be dominated by the penalty for known nonrelevant documents and becomes noisy.

[00108] Another example of an indicator which can be used in one or more embodiments is a coverage indicator, which can have one or more associated metrics that can be used as an estimate of an availability of results for a given query. Examples of coverage metrics include, without limitation, a “number of results”, a “number of hits”, and a “cache hit”. In accordance with one or more embodiments, a “coverage” parameter can be established, which represents a maximum number of result items in a result set. However, it is possible that a result set has fewer than the threshold maximum number of result items. The “number of hits” for a given result set represents the actual number of result items in the result set. The “number of results” can be expressed as a ratio of the “number of hits” to the threshold maximum number of result items. The “cache hit”, or “cache percentage”, comprises a percentage of the number of result items in a result set for a given crawl which have corresponding judgment input. On an aggregate level, the coverage metrics can be the average number of results over all queries for both crawls.

[00109] Examples of coverage metrics include, without limitation: 1) a percentage of queries in the QB with at least one result, 2) an average number of results per query in the QB, 3) a median number of results per query in the QB. A histogram can be used to show the absolute number of judgments for both the crawls at each position. Another metric can represent the number of empty queries(no results) for each crawl. If the cache hit ratio/percentage is 100%, judgment input is available for every result item. The relevance indicator, and performance estimation, can be considered to be highly accurate. If the cache hit ratio/percentage is less than 100%, performance estimation is likely less accurate. Embodiments of the present disclosure provide functionality for use in estimating performance in either case.

[00110] Performance test definition and evaluation system 211 can facilitate performance evaluation based on judgment input and using relevance and coverage indicators. In addition and using system 211, a user has an option to override judgment input. For example, a user can override judgment input if the user disagrees with the judgment input. In addition, system 211 provides various features to provide the user with increased flexibility in analyzing and testing performance, using one or more performance indicators (and corresponding one or more metrics), even in a case that a complete set of judgment input is unavailable. In accordance with one or more embodiments, system 211 allows the user to make “ad hoc” tuning selections/decisions, to include “past performance tests” in an analysis,

consider statistics concerning queries to be judged, and use absolute difference of scores, any two or more of which can be used in combination in accordance with one or more embodiments of the present disclosure.

[00111] In accordance with one or more embodiments, “ad hoc” tuning selections/decisions can comprise making decisions regarding relevance of results items for which judgment input is unavailable. For example, the user can indicate whether or not result items which have no associated judgment input are to be treated as relevant or nonrelevant result items. Alternatively, the user can indicate that precision/relevance is to be determined based on those result items which have associated judgment input, thereby excluding the result items for which judgment input is unavailable. As yet another option, which can be used, with one or more other “ad hoc” tuning selections/decisions, the user can elect to use a rank-based approach for determining precision/relevance.

[00112] System 211 further provides flexibility to use indicators/metrics corresponding to one or more other performance tests, in accordance with at least one embodiment. For example, the user can elect to perform one or more other precision determinations based on a precision determinations made in other performance tests. In accordance with at least one embodiment, queries, and their corresponding result sets, can be grouped based on the associated footrule and overlap metric values, i.e., the queries can be grouped based on whether or not a query returned the same results in the same order (footrule(0) and overlap(1)).

[00113] The queries can be separated into the queries with a footrule value of zero, i.e., those queries which had no change in their result sets from one crawl to the next, and those queries with a footrule value greater than zero, i.e., those queries that have differing results sets from one crawl to the next. The first category, or group, of queries (i.e., the group of queries whose footrule value is zero) is represented as N , and the second group of queries (i.e., the group of queries whose footrule value is other than zero) is represented as M . In addition, Z percent represents the precision calculation at a given position, or rank, in the result sets. The group, N , of queries whose footrule value is zero can be further divided into queries with complete judgment input, $N1$, and those queries with partial judgment input, $N2$, with the precision value for the $N1$ queries being X percent. For the remaining $N2$ queries with partial judgment input, the Z percent (which represents a precision calculation for a

given position, or rank) is used. The assumption is that the queries which return the same result for both crawls might still have the same precision. The combined precision value, X' for both subgroups of N , i.e., $N1$ and $N2$ can be determined as follows:

$$X' = (N1 * X1\%) + (N2 * Z\%)/(N1 + N2) \quad (25)$$

[00114] Turning to the group, M , of queries whose footrule value is other than zero, can be further divided into queries whose result set has complete judgment input, $M1$, and the group of queries whose result set has partial judgment input, $M2$. Let $Y1$ percent represent the precision calculation for the $M1$ group of queries. For the remaining $M2$ group of queries with partial judgment input, the precision can be extrapolated using the precision for complete judgment input at a given position or rank, or Z percent, as discussed above. The precision value, $Y2$, can be determined as follows:

$$Y2 = ((M1 * Y1\%) + (M2 * Z\%))/(M1 + M2) \quad (26)$$

[00115] The precision for the queries with different set of results can be determine as follows:

$$Y' = (M1 * Y1\%) + (M2 * Y2\%)/(M1 + M2) \quad (27)$$

[00116] The cumulative precision at a position can be determined as follows:

$$((N / M + N) * X') + ((M / M + N) * Y') \quad (28)$$

[00117] The cumulative precision at all positions is calculated similarly.

[00118] Precision can be computed over a differential set of queries. This analysis can be useful in a case that the data in providers/source 205 continuously changes, or changes at a greater frequency than other data sources. The new data might hold the same relevance as the previous data. This analysis can use a precision determination from judgment data associated with the result that is the same for both crawls. Precision, Pi can be calculated for a given position, i , in a result set, as follows:

$$Pi = (\text{Number of relevant results up to position "i" + change in the precision for the two crawls}) / \text{Number of results up to position i}$$

[00119] The change in precision at a position could be: $-I$, the result in a result set in the first crawl is relevant and the result at the same position in the same result set in second crawl

is irrelevant; 0, both the result at the same position in both of the first and second crawls have the same relevance, e.g., both are either relevant or irrelevant; or 1, the result in the first crawl is irrelevant and the result at the same position in the second crawl is relevant.

[00120] The change can be combined with a precision metric score determined for results up to the i th position in the ranking, or ordering, of the results. Thus, a total change shows either an improvement in performance (e.g., such as in a case that performance associated with the later crawl using a modified search engine 204 is considered to be improved over performance associated with a previous crawl using the same search engine 204 without the changes). For the remaining set of queries which haven't changed, the previously-determined precision at that position is assumed. For this analysis, it is possible to only analyze those results which differ at a particular position out of all the queries which have footrule > 0 (queries with different results). It is a quick and simple method to analyze the changes in the results.

[00121] In accordance with embodiments of the present disclosure, system 211 allows the user to determine, for a given performance test, for example, the queries for which judgment input has not been received, and/or the number of results per query for which judgment input has not been received, and overlap and footrule distance measures. Embodiments provide the user with an option to estimate the remaining judgments. For example, the user has an option to generate one or more estimates for: all of the queries for which judgment input is not available, the queries with a footrule >0 (e.g., those queries that differ in their results or in the order of the results) and/or queries with partial judgment input, and/or queries with non-matching results at a position for which there is no judgment input.

[00122] In accordance with embodiments of the present disclosure, system 211 can provide a histogram, which represents absolute difference of scores at each position, such as the absolute gain and loss (e.g., precision gain or loss) at each position over the entire set of queries, which can represent performance from one crawl to the next. The number of pairs that were compared to generate the absolute number can also be provided with the histogram; in order to facilitate analysis of the histogram.

[00123] In accordance with one or more embodiments, an estimate can be provided before a performance test commences using system 211. For example, judgment input with a consistent scale for a series of performance test can be aggregated using a relevance scale

corresponding to the previous performance tests. The judgment input with a consistent scale for a series of performance tests can be aggregated using a relevance scale. These judgments can be used to estimate a precision for the current performance test using the same scale as of the aggregated judgments. The current performance test needs to be setup or at least the crawl for the providers needs to be complete in order to configure the pre-performance test analysis. The information used to specify the crawls for the new performance test can then be used perform the pre-performance test. In accordance with at least one embodiment, one or more precision curves can be displayed at each grade on the relevance scale for every position. Two kinds of precision curves can be generated, for example, precision for all the result items having no judgment input, and the other for result items which have judgment input.

[00124] Performance test definition and evaluation system 211 enables a user to analyze various aspects of search results, including variations in search results, in relevance of results or in the coverage. System 211 provides an ability to analyze a search engine's 204 performance, at one or more levels, e.g., at query/result level or an aggregate level. The following provides an example of output that can be provided by system 211, and possible interpretations of the output, in accordance with one or more embodiments of the present disclosure. It should be apparent that other types of output and interpretations are possible.

[00125] System 211 can provide an aggregate relevance (or precision) value to compare two crawls (e.g., which crawls can involve two instances of search engine 204), which relevance can be obtained from a relevance tab of a user interface provided by system 211, after selecting a given performance test. In addition, a simple average cumulative precision across all the queries and DCG can be provided at both a query level and an aggregate level. A simple average cumulative precision can be computed across all queries in a QB corresponding to a performance test, and a graph can be provided which illustrate both crawls. The curves shown in the graph can be low (e.g., assuming all results without judgments are irrelevant), and high (e.g., assuming all the results without judgments are relevant), an average precision (average of the low and high curves) and/or a predicted precision from the previously-conducted performance tests.

[00126] An actual precision can lie somewhere between the low and high curves. If the range of the two curves is close, then it can be assumed that a reasonable estimate of the

precision can be made. In addition and if the range of the high and low curves is larger, confidence that an estimate reflects an actual precision is lower. Confidence can also be based on cache hit. A high cache hit ratio implies that a good prediction of the precision could be obtained and vice versa. Experimental validations have shown that at least a forty to fifty percent cache hit can be sufficient to determine whether one configuration of a search engine 204 is performing better or worse than another search engine 204 configuration, for example.

[00127] An average precision curve can use the average of the low and high curves using an assumption of a 50% probability. The rank-based predicted precision curve generates a precision curve based on precision results of previous performance tests. One assumption is that a subsequent search engine 204 configuration may not have a worse configuration than a previous search engine 204 configuration. In such a case, an average of precision results from one or more previous performance tests for the previous search engine 204 configuration can be used to compute precision results for the subsequent search engine 204 configuration, which can be a function of the low and high precision curves. The following illustrates one example of a rank-level precision calculation, which can be used in connection with one or more embodiments, where i represents a relevance rank in a result set:

$$\text{Rank predicted precision (i)} = \text{low precision(i)} + ((\text{high precision(i)} - \text{low precision(i)}) * \text{prior average precision for one or more previous tests(i)})$$

[00128] The prior precision values can be obtained from analysis database 210, for example. Judgement database 206 can include minimum, maximum, average and recent precision scores for one or more previous performance tests. In accordance with one or more embodiments, the user can specify the previous performance tests to use to estimate precision, or a default specification (e.g., all performance tests for a given search engine 204 configuration) can be used to build a predictive model.

[00129] In accordance with one or more embodiments, a DCG can be computed, and one or more graphs can be displayed to show, for both crawls in a performance test, a low and high DCG curve, i.e., corresponding to assumptions that all unknown judgments either irrelevant and relevant, respectively. The DCG can also be computed on the query level.

[00130] A distance metric (e.g., one or all of Footrule, Kendall Tau, and overlap metrics) can be used, in accordance with one or more embodiments, to provide a measure of the extent

of overlap in the results between the two crawls. Footrule, Kendall and overlap metrics are computed.

[00131] On an aggregate level, cumulative match and pie chart graphs can be output showing a footrule distribution. Figures 5A and 5B illustrate pie chart and cumulative match graphs for use with one or more embodiments of the present disclosure. Referring to Figure 5A, a pie chart is illustrated which shows a percentage of the queries with a distance change breakdown, which shows a distribution of the distance change. The distance change is given in the range of zero to one, with 0 indicating that the results of the two crawls (e.g., and the two configuration of search engine 204) being compared are the same, and 1, at the other end of the spectrum, indicating that the results of the two crawls are very different or the results are the same but are ranked in a different or reverse order. For example, section 501 of the pie chart corresponds to a zero change between the two crawls, and indicates that fifty percent of the result sets were unchanged between the crawls.

[00132] The cumulative match chart can be used to show a number of results that overlap up to a ranking or position. The cumulative match chart can be used to evaluate a percentage of results that overlap in the two result sets corresponding to the two crawls. In accordance with one or more embodiments, the cumulative match data is cumulative across each rank, or position, and is aggregated over all the queries in the performance test. A high cumulative match percentage for a given position indicates that the results in the crawls are predominantly the same up to that position. Figure 5B provides an example of a cumulative match chart, which shows a percentage of overlap by position, or rank.

[00133] On the query level, the footrule and kendall metrics for each query can indicate a difference in the results themselves and/or in the ordering of the results. A high value of footrule or kendall indicates the results in the two crawls are ranked differently and/or each engine returned a different set of results. A footrule or kendall value of zero can indicate that the results of the two crawls returned the same result set. A user's examination can thereby be focused on the one or more queries that have an associated high footrule and/or kendall metric value, in order to identify the differences in the results, and/or to gain an understanding of the reasons for the differences. An overlap measure can be used to indicate the distribution of results in the two crawls without taking into account the ranking of the result. A range can be between zero and one, with 0 indicating those queries that have no

overlap in results sets between the two crawls, and I , and the other end of the spectrum, indicating a 100% overlap in the results between the two crawls. Embodiments of the present disclosure can sort the queries in accordance with their overlap metric value, so that the user can identify the queries for which the results sets are different, in order to examine them more closely (e.g., to identify the differences and or the reasons for the differences).

[00134] In accordance with at least one disclosed embodiment, coverage metrics can be provided on a query level and/or at an aggregate level. On an aggregate level, a number of results returned per query can be aggregated across an entire result set. The aggregate number can be used to determine which of the two crawls has the higher coverage, for example. The absolute number aggregated across an entire result set can be determined for each result position (i.e., a number one ranked result, number two ranked, etc.) and each crawl, and the aggregate number for each position and each crawl can be displayed in a histogram, with the two numbers corresponding to the two crawls for a given position being represented side-by-side in the histogram. The user can then review the histogram to evaluate coverage issues by result position, for example.

[00135] A cache hit can be provided, in accordance with one or more embodiments, which cache hit can be represented as a percent of the number of judgments found for the queries. A high cache hit suggests that the performance test provides a good prediction or estimate of performance, for example. On a query level, a cache hit can be reported as a number of results, the number of hits and a percentage of judgment input available per query. The number of results reported on the query level can be used, for example, to identify queries that are returning a different number of results in the two crawls.

[00136] Figure 5C provides an example of an aggregate precision metric, which identifies a percent of result items at a given relevance rank found to be relevant. In other words and with respect to the first crawl, 76.32% of the judgment input indicated that a result item ranked the most relevant in a query result set was relevant. The value is aggregated across all of the queries in the first crawl. In contrast, 75.95% of the judgment input corresponding to the second crawl indicated that the number one ranked relevant result item for the queries in the second crawl was relevant.

[00137] The following provides an example of a process that can be used to analyze a performance test and its corresponding results in accordance with one or more embodiments. The example assumes that a reasonable number of judgments are available.

[00138] An initial check of the cache hit percentage can be made to determine a degree of coverage. If there is an acceptable degree of coverage for each of the two crawls, an average precision curve can be examined as well as a predicted precision curve, which is based on previously-conducted performance tests. An average precision curve shows a high curve, which corresponds to an assumption that missing judgments are irrelevant, and a low curve, which corresponds to an assumption that missing judgments are relevant. Figure 5D provides an example of average precision curves 542 for the two crawls and of rank-based prediction curves 540 in accordance with one or more embodiments. As discussed herein, a minimal difference between the curves suggests that a reasonable estimate of the precision can be made. A predicted precision can be, for example a rank-based prediction. The precision curves for the two crawls can be compared to evaluate which one, if any, of the crawls indicates a better performance.

[00139] Aggregate distance metrics can be examined to determine whether or not the two crawls have a high percentage of overlap, which can suggest minimal change in the results. If there is not a high percentage of overlap, it can be assumed that there is a performance issue. In such a case, the queries that have a high degree of variance between the two crawls can be examined in order to determine which crawl has a better associated performance. With regard to the queries that have a high variance, system 211 can compute an adhoc relevance that have associated judgment input, as one indicator of performance of the two crawls.

[00140] In a case that there is insufficient judgment input for a performance test, a distance between the two crawls can be estimated. If there is a high variance, its difficult to construct a strong proposition of relevance, but a small number of highly varying queries could be spot checked and judged and a quick estimate of the two crawls could be obtained. If the number of queries that vary between the two crawls is minimal, a spot check can be performed on the queries based on an overlap and/or on a footrule, and the queries can be randomly judged.

[00141] Figure 6 provides an example of a database structure of analysis database 210 in accordance with one or more embodiments of the present disclosure. In one or more

embodiments, analysis database 210 includes tables 601 to 614, which contain information corresponding to queries, query results, judgment input, QBs, performance test, etc. Query 601, for example, contains information concerning each of the queries. A “QB2Query” table can be used to associate multiple queries to a QB, and table 602. In the example shown, a QB can include multiple queries, and a query can be included in multiple QBs. Table 605 corresponds to a performance test instance, and includes a relational reference, “qbid”, to the QB table 602. The provider table 608 corresponds to crawl instances, and table 611 associates a crawl to a performance test. Table 606 provides a rank for a result and associates the ranked result (table 604) to a crawl (table 608), query (table 601) and test (table 605).

[00142] A result stored in result table 604 is associated with a result value stored in table 613. Tables 607, 612 and 614 store “side-by-side”, “per set”, “per item” judgment input, respectively, for results stored in result table. In addition, table 610 stores one or more questions associated with judgment input, e.g., a question used by judgment system 207 to prompt a judge 208 to enter the judgment input.

[00143] Those skilled in the art will recognize that the methods and systems of the present disclosure may be implemented in many manners and as such are not to be limited by the foregoing exemplary embodiments and examples. In other words, functional elements being performed by a single or multiple components, in various combinations of hardware and software or firmware, and individual functions, can be distributed among software applications at either the client or server level or both. In this regard, any number of the features of the different embodiments described herein may be combined into single or multiple embodiments, and alternate embodiments having fewer than or more than all of the features herein described are possible. Functionality may also be, in whole or in part, distributed among multiple components, in manners now known or to become known. Thus, myriad software/hardware/firmware combinations are possible in achieving the functions, features, interfaces and preferences described herein. Moreover, the scope of the present disclosure covers conventionally known manners for carrying out the described features and functions and interfaces, and those variations and modifications that may be made to the hardware or software or firmware components described herein as would be understood by those skilled in the art now and hereafter.

CLAIMS

We claim:

1. An information retrieval engine evaluation method, comprising:
identifying a query benchmark comprising a plurality of queries, the queries
having corresponding search results;
obtaining judgment input from one or more judges, the judgment input
corresponding to the set of search results;
determining, using the judgment input obtained from the one or more judges,
at least one metric corresponding to an indicator of performance, a first
value of the at least one metric corresponding to a first information
retrieval engine and a second value of the at least one metric
corresponding to a second information retrieval engine; and
comparing the first and second values of the at least one metric to evaluate the
performance indicator so as to evaluate performance of the first
information retrieval engine relative to the second information retrieval
engine based on the query benchmark.
2. The method of claim 1, wherein the first and information retrieval engines
comprise search engine instances and the performance indicator comprises an indicator of
relevance of search results generated by the search engine instances.
3. The method of claim 1, further comprising
determining at least one other metric corresponding to at least one other
indicator of performance, a first value of the at least one other metric
corresponding to the first information retrieval engine and a second
value of the at least one other metric corresponding to the second
information retrieval engine; and
comparing the first and second values of the at least one other metric to
evaluate the at least one other performance indicator so as to evaluate
performance of the first information retrieval engine relative to the
second information retrieval engine.

4. The method of claim 3, wherein the first and second information retrieval engines comprise search engine instances and the at least one other performance indicator comprises an indicator of differences between search results generated by the search engine instances.
5. The method of claim 3, wherein the first and second information retrieval engine comprise search engine instances and the at least one other performance indicator comprises an indicator of coverage of search results generated by the search engine instances.
6. The method of claim 1, wherein the first and second information retrieval engines comprise different configurations of a search engine.
7. The method of claim 1, wherein the first and second information retrieval engine comprise different search engines.
8. The method of claim 1, further comprising:
examining the judgment input to identify inconsistencies in the judgment input.
9. The method of claim 8, wherein examining the judgment input further comprises:
examining the judgment input to identify an inconsistency in a given judge's judgment input.
10. The method of claim 8, wherein examining the judgment input further comprises:
examining the judgment input to identify an inconsistency in one judge's judgment input relative to at least one other judge's judgment input.
11. A method of measuring search engine performance using a set of stored queries, results and judgments, the method comprising:

generating a query benchmark from the set of stored queries, the query benchmark includes one or more queries;
obtaining query results using the one or more benchmark queries;
retrieving one or more stored results associated with the plurality of queries;
retrieving a judgment associated with at least one stored result;
predicting a judgment associated with one or more of the obtained results based on the at least one stored result; and
determining a performance measure using the retrieved and predicted judgments.

12. The method of claim 11, wherein a judgment associated with a stored result identifies relevancy of the stored result to the query.

13. The method of claim 11, further comprising:
storing each of the plurality of queries in a query log;
for each query stored in the query log:
performing the query to obtain a set of results, the set of results including at least one result item; and
obtaining a judgment corresponding to the at least one result item.

14. The method of claim 13, further comprising:
examining an obtained judgment to determine whether an inconsistency exists in the obtained judgment.

15. The method of claim 14, wherein a judgment is based on user input, and wherein examining an obtained judgment to determine whether an inconsistency exists in the obtained judgment further comprises:

comparing a user's judgment input associated with a result item to predetermined judgment input for the associated result item.

16. An information retrieval engine evaluation system, comprising:

program memory for storing process steps executable to: identify a query benchmark comprising a plurality of queries, the queries having corresponding search results, obtain judgment input from one or more judges, the judgment input corresponding to the set of search results determine, using the judgment input obtained from the one or more judges, at least one metric corresponding to an indicator of performance, a first value of the at least one metric corresponding to a first information retrieval engine and a second value of the at least one metric corresponding to a second information retrieval engine; and compare the first and second values of the at least one metric to evaluate the performance indicator so as to evaluate performance of the first information retrieval engine relative to the second information retrieval engine; and

at least one processor for executing the process steps stored in said program memory.

17. A system for measuring search engine performance using a set of stored queries, results and judgments, the system comprising:
- program memory for storing process steps executable to: generate a query benchmark from the set of stored queries, the query benchmark includes one or more queries; obtain query results using the one or more benchmark queries; retrieve one or more stored results associated with the plurality of queries; retrieve a judgment associated with at least one stored result; predict a judgment associated with one or more of the obtained results based on the at least one stored result; and determine a performance measure using the retrieved and predicted judgments; and
- at least one processor for executing the process steps stored in said program memory.

1/8

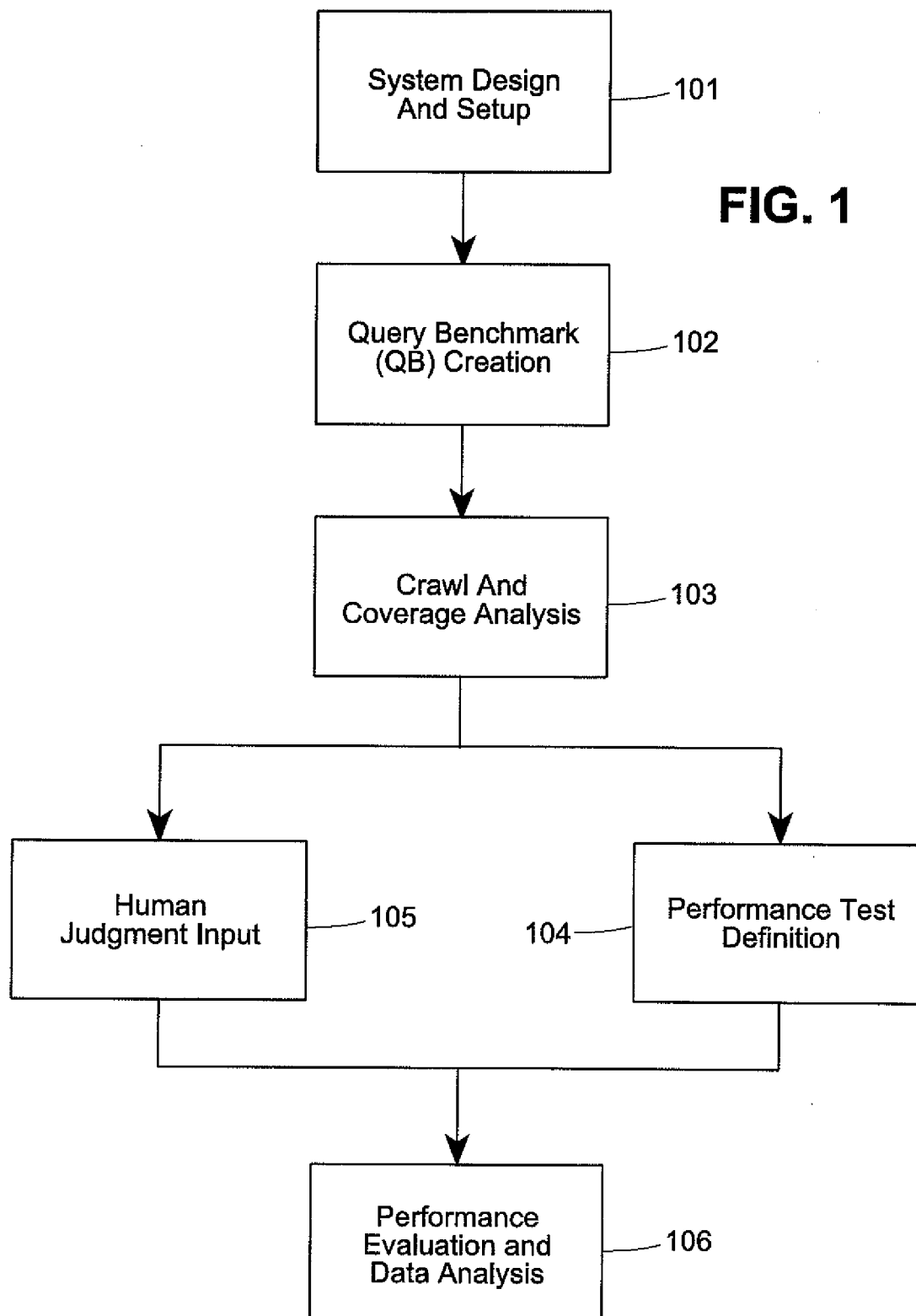
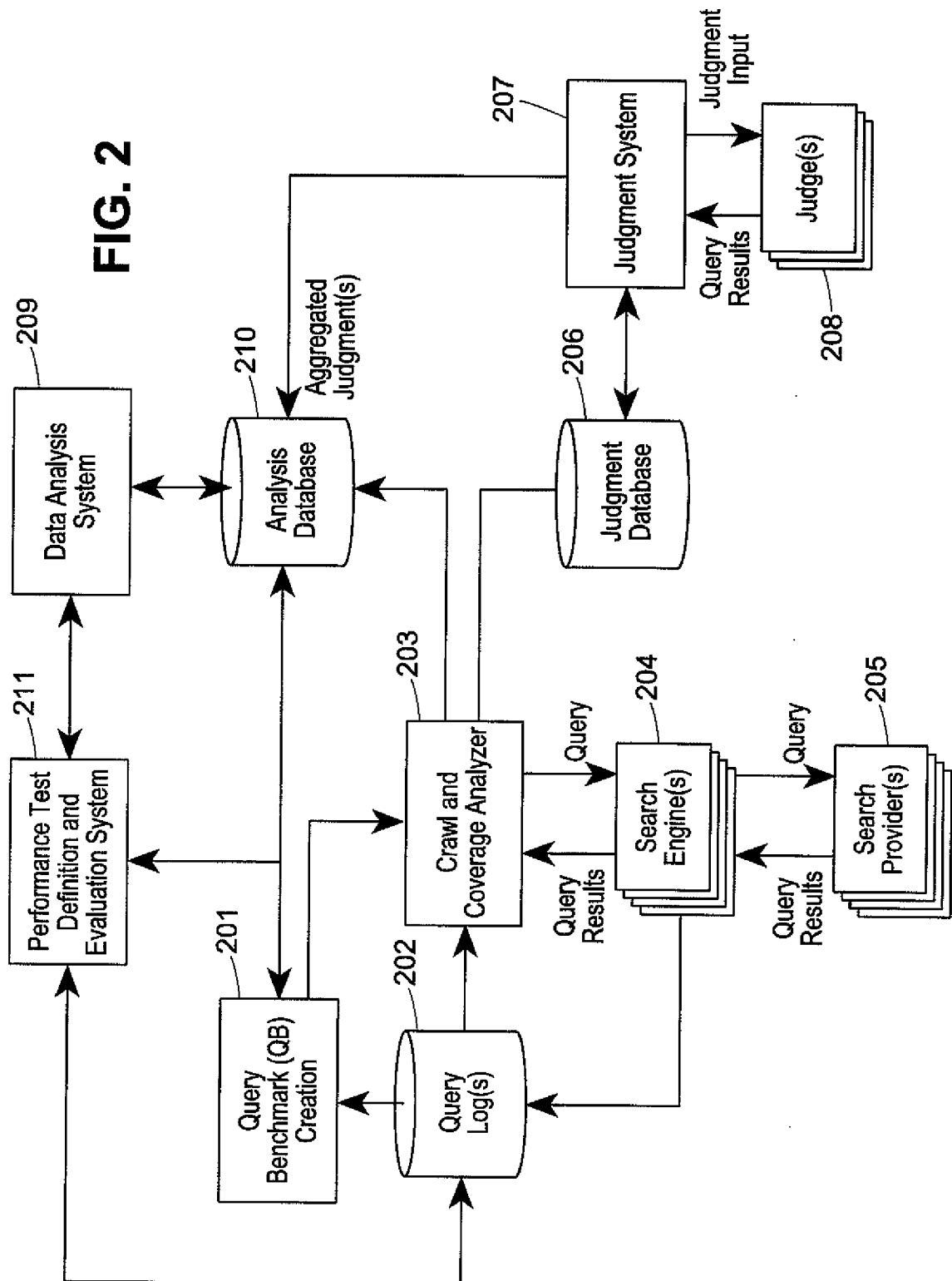


FIG. 2



302

256 MB Module For XYZ Computer System

312

Supercharge your computer system with a ABC memory upgrade...

322

256 MB Module computer system memory upgrade—301

303

305

How relevant does this SEARCH RESULT appear to be for the query?

306

Accept	☒
Conditional Accept	☐
Reject - Not Relevant	☐
Other Reject	☐
No Judgment	☐

307

How relevant is this search result's LANDING PAGE for the query?

308

Accept	☒
Conditional Accept	☐
Reject - Not Relevant	☐
Other Reject	☐
No Judgment	☐

FIG. 3

4/8

Spearman's Footrule Distance: $F(R1, R2)$

Determine distance between the permutations as vectors.

Example:

401	402	
A	B	
B	D	shift(A) = 2 shift(B) = 1, shift(C) = 1,
C	A	and shift(D) = 2, $F(R1, R2)$ is 6
D	C	

FIG. 4A**Kendall-tau Distance $K(R1, R2)$**

Number of pairs in which the items in R1 and R2 disagree on.

Example:

401	402
A	B
B	D
C	A
D	C

Number of disagreements: $K(R1, R2)$ is 3 $\rightarrow$ (AB, AD, CD)**FIG. 4B**

5/8

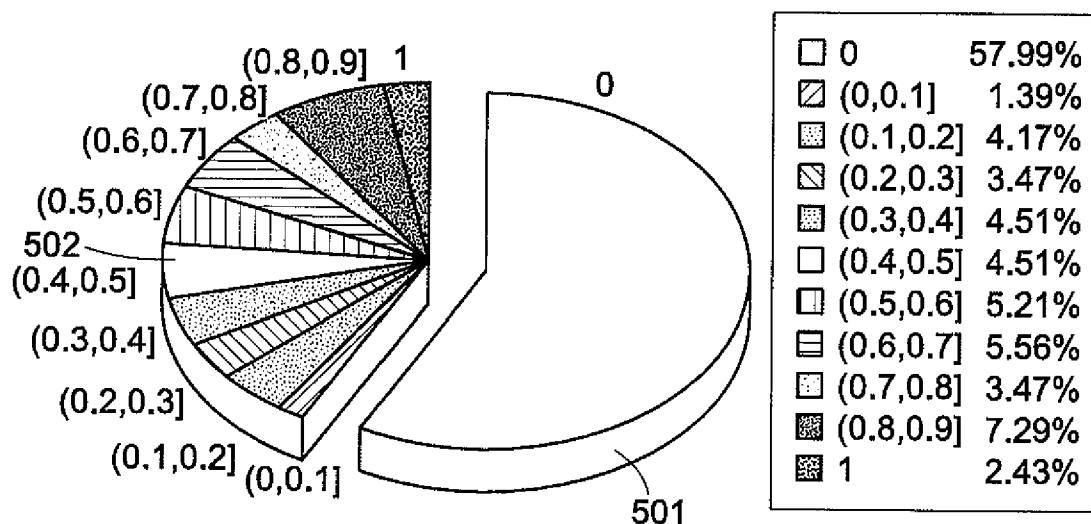
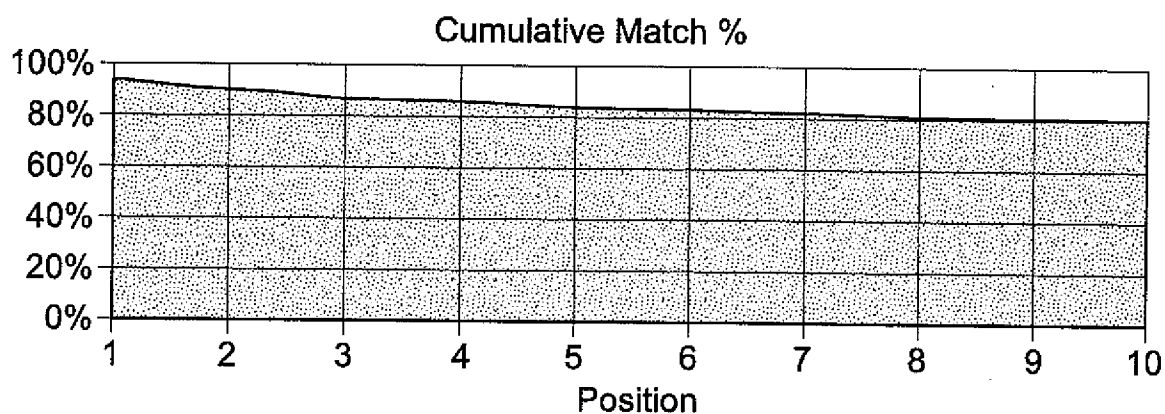


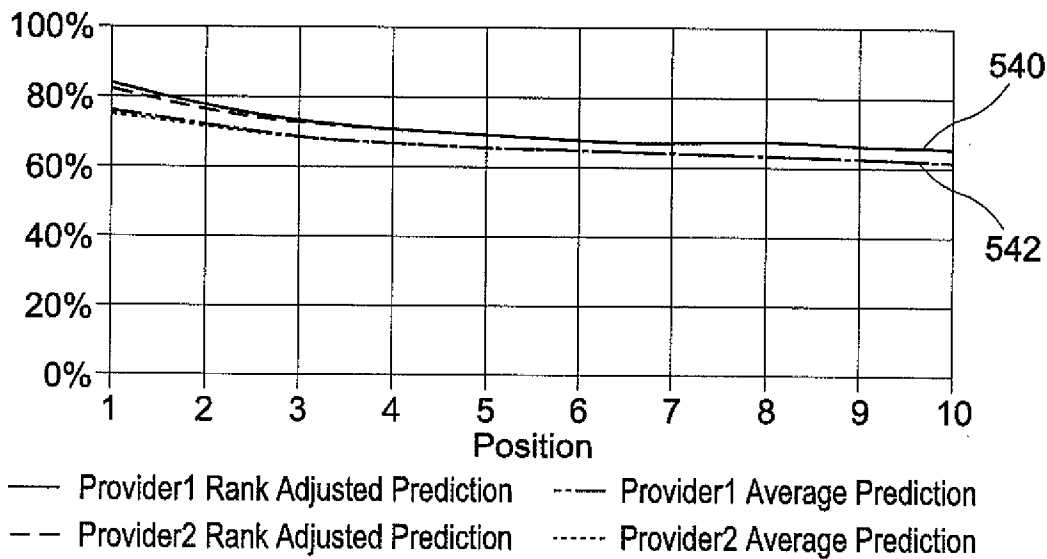
FIG. 5A

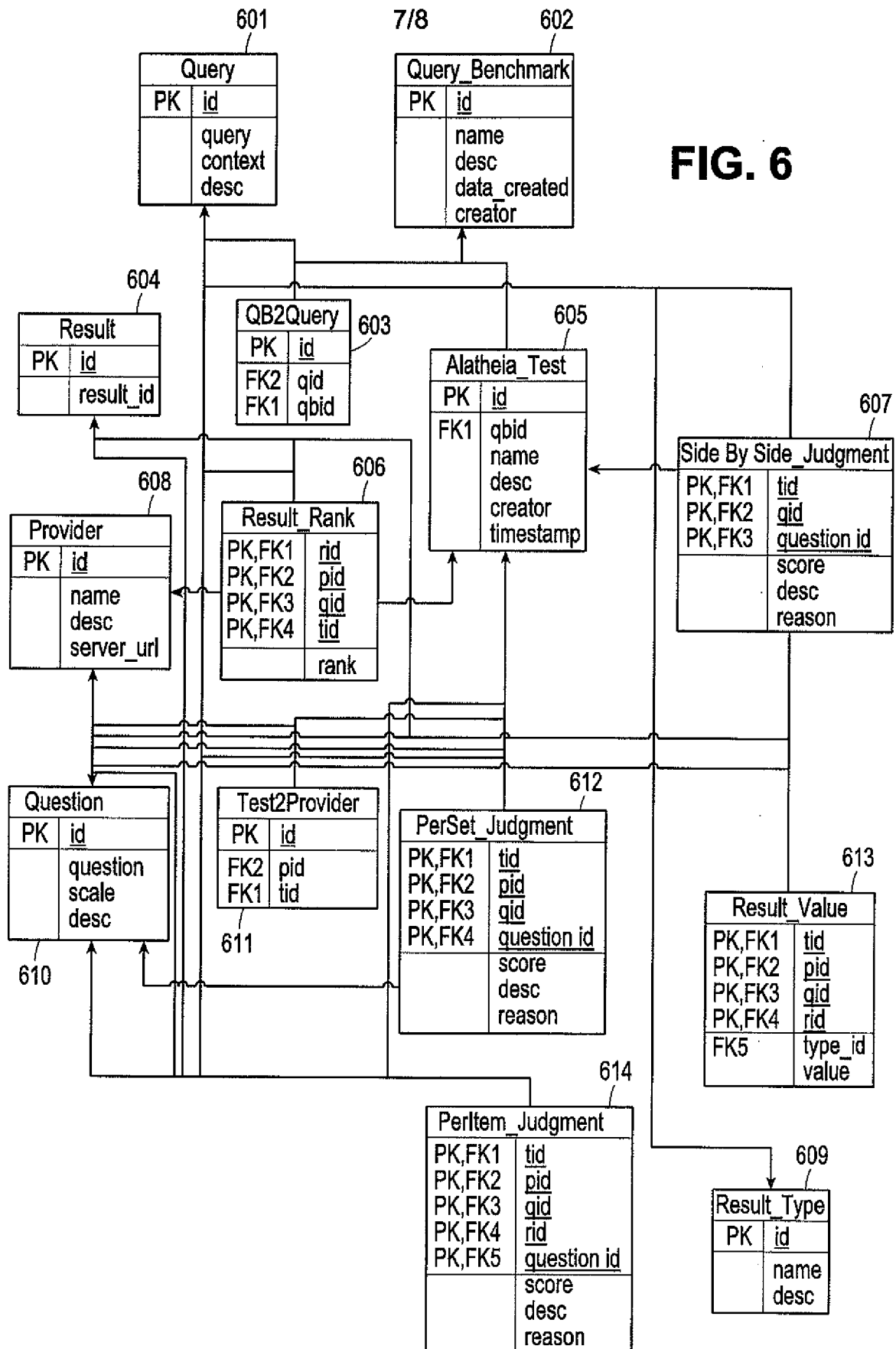
FIG. 5B



6/8

Position	Crawl No. One	Crawl No. Two
1	76.32%	75.95%
2	69.62%	72.78%
3	69.39%	73.11%
4	68.83%	72.39%
5	67.07%	70.99%
6	66.65%	71.15%
7	67.27%	71.28%
8	67.69%	70.56%
9	67.52%	70.45%
10	67.55%	70.85%
11	67.47%	70.75%
12	67.71%	70.63%
13	67.55%	70.22%
14	67.84%	70.08%
15	67.58%	70.01%

FIG. 5C**FIG. 5D**



8/8

700

704		Model		Infit		708
Judge No.	Measure	S.E.	Mn Sq	ZStd		
702	20	-1.21	0.1	0.46	-6.4	
	14	-1.19	0.1	1.07	0.7	
	29	-1.17	0.1	0.82	-1.8	
	5	-0.78	0.1	0.73	-2.6	
	7	-0.57	0.1	0.51	-5.3	
	15	-0.5	0.1	0.82	-1.7	
	32	-0.46	0.1	1.19	1.6	
	1	-0.37	0.11	1.29	2.3	
	28	-0.36	0.11	0.52	-5	
	13	-0.35	0.11	1.5	3.8	
706	33	-0.28	0.11	0.92	-0.7	
	4	-0.27	0.11	0.77	-2.1	
	27	-0.25	0.11	2.95	9	
	25	-0.15	0.11	0.76	-2.2	
	9	-0.14	0.11	0.71	-2.6	
	24	-0.06	0.11	0.83	-1.4	
	10	-0.01	0.11	0.81	-1.6	
	2	0.09	0.11	1.21	1.6	
	6	0.11	0.11	0.99	0	
	3	0.16	0.11	0.79	-1.8	
	21	0.16	0.11	1.42	3	
	19	0.18	0.12	1.17	1.3	
	16	0.21	0.12	0.99	0	
	17	0.29	0.12	1.04	0.3	
	8	0.4	0.12	0.99	0	
	34	0.4	0.12	0.67	-2.8	
	11	0.48	0.12	0.96	-0.2	
	23	0.5	0.12	0.95	-0.3	
	30	0.55	0.12	1.09	0.7	
	31	0.7	0.13	0.91	-0.6	
	18	0.82	0.13	0.8	-1.5	
	26	0.93	0.13	1.76	4.5	
	22	0.95	0.13	0.74	-2	
	12	1.18	0.14	1.2	1.3	
Average		0	0.11	1.01	-0.4	
Std. Dev.		0.59	0.01	0.44	2.9	

FIG. 7

A. CLASSIFICATION OF SUBJECT MATTER**G06Q 99/00(2006.01)i, G06F 17/30(2006.01)i**

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 8 G06F 17/00, 17/30, G06Q

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

Korean Utility models and applications for Utility Models since 1975

Japanese Utility models and applications for Utility Models since 1975

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

eKIPASS(KIPO Internal) & keyword : (search, retrieval) engine, performance evaluation and similar terms

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 6,877,034 B1 (Machin et al.) 05 April 2005 see abstract; column 1 line 14 - column 2 line 50; claims 1-3.	1-17
A	US 2005/0154716 A1 (Watson et al.) 14 July 2005 see abstract; figures 1-3; claims 1-10; paragraphs 0008-0013.	1-17
A	US 2004/0220915 A1 (Kline et al.) 04 November 2004 see figures 1-2; claims 1-10.	1-17
A	KR 10-2001-0034993 A (HEO, NO JAE) 07 May 2001 see claims 1-3; figures 3-4.	1-17
A	US 2004/0220914 A1 (Chenng et al.) 04 November 2004 see abstract; paragraphs 0002-0018.	1-17



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

26 DECEMBER 2007 (26.12.2007)

Date of mailing of the international search report

26 DECEMBER 2007 (26.12.2007)

Name and mailing address of the ISA/KR

Korean Intellectual Property Office
920 Dunsan-dong, Seo-gu, Daejeon 302-701,
Republic of Korea

Facsimile No. 82-42-472-7140

Authorized officer

SON, Young Tae

Telephone No. 82-42-481-5748



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2007/074224

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US6877034B1	05.04.2005	None .	
US20050154716A1	14.07.2005	BR200501151A CA2501668AA CA2501668A1 CN1677403A EP01580670A1 EP1580670A1 JP17276213 JP2005276213A2 KR1020060044525 KR2006044525A MXPA05003141A US2005154716A1 US2005154716AA	01.11.2005 22.09.2005 22.09.2005 05.10.2005 28.09.2005 28.09.2005 06.10.2005 06.10.2005 16.05.2006 16.05.2006 05.10.2005 14.07.2005 14.07.2005
US20040220915A1	04.11.2004	CN1816810A EP01620818A1 EP1620818A1 JP18525603 JP2006525603T2 KR1020060017767 KR2006017767A US2004220915A1 US2004220915AA W02004100046A1 W02004100046C1	09.08.2006 01.02.2006 01.02.2006 09.11.2006 09.11.2006 27.02.2006 27.02.2006 04.11.2004 04.11.2004 18.11.2004 31.03.2005
KR1020010034993	07.05.2001	None .	
US20040220914A1	04.11.2004	CN1784679A EP01620819A1 EP1620819A1 JP18525604 JP2006525604T2 KR1020060030020 KR2006030020A US2004220914A1 US2004220914AA US2005065928A1 US2005065928AA W02004100022A1 W02004100022C2	07.06.2006 01.02.2006 01.02.2006 09.11.2006 09.11.2006 07.04.2006 07.04.2006 04.11.2004 04.11.2004 24.03.2005 24.03.2005 18.11.2004 07.07.2005