A system and method receives identification of a web page. The web page is added to a web page library. Upon occurrence of an analysis event, the web page is provided to an automated analysis framework. Analyses representing SEO best practices are run against the web page. Each of the analyses is associated with input data.
The following results indicate how closely your web page addresses items that increase your natural search engine rankings: 3% of the analyses failed, 14% of the analyses need attention and 83% of the analyses passed.

- Primary keyword in meta description tag
- Secondary keywords in meta keywords tag
- Existence of keywords in content
- Registered with all major search engines
- Age of site
- Page size
- No broken links

**FIGURE 4**
Web Page: Powerful, automated search engine marketing | Yield Software
URL: http://www.yieldsoftware.com
Analysis: Primary keyword in meta description tag

Description
Many search engines display the meta description tag in their search results, especially if the term that the user searched for is found in the meta description tag. In addition some search engines may use the words in your meta description tag to classify your page if they are unable to extract meaningful content from the rest of your page.

Recommendation
Change your meta description tag to include your primary keyword (yield software) and variations (e.g. synonyms, etc.). You can also include secondary keywords (pay per click, seo software, web marketing, web marketing made easy). The tag should be no more than a couple of sentences and should read well. As visitors may see this tag, it is important to provide a meta description that is compelling and on topic so that you can both increase your click-through rates and increase conversions once traffic arrives at your site.

History

FIGURE 5
Recommendation

Many search engines display the meta description tag in their search results, especially if the term the user searched is found in the meta description tag. In addition, some search engines may use the words in your meta description tag to classify your page if they are unable to extract meaningful content from the rest of your page.

Modify your meta description tag to include your target keywords and variations (e.g. synonyms, etc.).

The tag should be no more than a couple of sentences and should read well. Since searchers may see this tag, it is important to make it compelling and on-topic.

Enter your meta description tag into the field below.

Yield is a great search engine marketing (SEM) solution.

Unique meta description tag within site
METHOD AND SYSTEM FOR AUTOMATED SEARCH ENGINE OPTIMIZATION

RELATED APPLICATIONS

This application claims priority to provisional U.S. Patent Application Ser. No. 61/088,966, filed on Aug. 14, 2008 and entitled “Method and System for Optimization, Automation, and Administration”, the disclosure of which is hereby incorporated herein by reference.

FIELD OF THE INVENTION

The present application relates to Internet searching and, in particular, a method and system for analyzing a web page for conformance to search engine optimization best practices and fixing or otherwise modifying the web page to improve search engine rankings.

BACKGROUND

Many persons use the Internet to make purchases. In the purchasing process, users may employ search engines to find desired products and services. The user enters one or more words, called a search term, into the search engine. The search engines display the web pages that best match the users search terms on the search engine results page (SERP) in a region commonly called the organic or natural search results. The higher the web page ranks for a particular search term, the more people typically click to view the web page. Thus, many businesses try to get their web pages to rank as high as possible on the SERP for many search terms.

In a process called “spidering”, each search engine reads most of the pages on the internet. Then, based on an internal scoring algorithm, the search engine ranks (or calculates in real-time upon a user’s query) each web page’s relevance for each search term. The ranking is based on a variety of criteria, for example, the existence of keywords in the content of the page, the internal structure of the page, the number and quality of other pages linking to the page, and a variety of other criteria.

The practice of modifying a web page as well as additional content on the internet (e.g., creating links from third-party sites, defining the anchor text or text displayed on those links, etc.) to drive the web page higher in the SERP is called search engine optimization (SEO).

Some of the criteria used for ranking pages are public information, while others are non-public and only known by experts at search engine companies. Additionally, the criteria change over time. Many individuals in the field continually theorize about what criteria may be used by the search engine’s relevance ranking. It may not be publicly known whether these theoretical criteria are used by the search engine. However, if many individuals believe they may have an impact, theoretical criteria, together with the criteria made public by the search engines, may comprise the “best practices” for search engine optimization.

SUMMARY

In one embodiment, a system and method receives identification of a web page. The web page is added to a web page library. Upon occurrence of an analysis event, the web page is provided to an automated analysis framework. Analyses representing SEO best practices are run against the web page. Each of the analyses is associated with input data.

BRIEF DESCRIPTION OF THE DRAWINGS

The present disclosure is illustrated by way of example and not limited in the figures of the accompanying drawings in which like references indicate similar elements.

FIG. 1 illustrates a system in accordance with one embodiment of the present invention.

FIG. 2 illustrates a functional block diagram and method in accordance with one embodiment of the present invention.

FIG. 3 illustrates a functional block diagram and method in accordance with one embodiment of the present invention.

FIG. 4 illustrates a screen in accordance with one embodiment of the present invention.

FIG. 5 illustrates a screen in accordance with one embodiment of the present invention.

FIG. 6 illustrates a functional block diagram and method in accordance with one embodiment of the present invention.

FIG. 7 illustrates a screen in accordance with one embodiment of the present invention.

FIG. 8 illustrates a block diagram of one embodiment of a system in accordance with one embodiment of the present invention.

FIG. 9 shows a diagrammatic representation of a machine in an exemplary form of a computer system in accordance with one embodiment of the present invention.

DETAILED DESCRIPTION

In the following description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the description. It will be apparent, however, to one skilled in the art that embodiments of the disclosure can be practiced without these specific details. In some instances, structures and devices are shown in block diagram form in order to avoid obscuring the description. In other instances, functional block diagrams are shown to represent data and logic flows.

Reference in this specification to “one embodiment”, “an embodiment”, “other embodiments”, or the like means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the disclosure. The appearances of, for example, the phrase “in one embodiment” in various places in the specification are not necessarily all referring to the same embodiment, nor are separate or alternative embodiments mutually exclusive of other embodiments.

Moreover, whether or not there is express reference to an “embodiment” or the like, various features are described which may be variously combined and included in some embodiments but also variously omitted in other embodiments. Similarly, various requirements are described which may be requirements for some embodiments but not other embodiments.

The present invention is a system and method of automated testing of web pages and landing pages (collectively called “web pages”) for compliance with best practices (also known as “analyses”) for search engine optimization. The present invention is also a system and method of auto-
mating fixes or modifications to web pages and landing pages to realize best practices for search engine optimization.

[0023] FIG. 1 illustrates a system 100 of analyzing a web page for conformance with best practices using automated search engine optimization (SEO) and fixing the web page or content on the internet to improve search engine rankings in accordance with the present invention. In one embodiment, the present invention can be used in connection with a variety of search engines. As used herein, the term “fix” or “fixing” with respect to a web page means modifying or otherwise optimizing the web page to improve search engine rankings. In another embodiment, the present invention can be used in connection with a variety of content management systems.

[0024] The system 100 includes user computing devices 102, Internet 108, search engines 104, web sites 106, and a platform 110. The platform 110 includes web servers 112, an analytics server 114, business logic server 116, and database server 118. A user can access the Internet 108 via user computing devices 102 and use the functionality of the platform 110. The user can then specify web pages to be optimized which are routed through the server 12 to the business logic server 116 and stored in the database server 118. The business logic server 116 performs the automated search engine optimization by retrieving information from search engines 104 and web sites 106 via the Internet 108 and the web servers 112. Both the business logic server 116 and the analytics server 114 retrieve, process and store information in the database server 118. Then, the business logic server 116 conducts the automated SEO analysis and through the web server 112, Internet 108 and user computing devices 102 interacts with the user to implement a manual or auto-fix of the web page.

[0025] FIG. 2 illustrates a functional block diagram and method 200 of analyzing a web page for conformance to search engine optimization best practices and fixing the web page to improve search engine rankings in accordance with one embodiment of the present invention. The method 200 includes analysis library 202, web page library 204, database server 206, analysis scheduler 208, analysis queue 210, automated analysis framework 212, Internet 214, web page cache 216, cache manager 218, and content management server 220. Also shown is a user interface 222 of the user computing devices 102. The user interface 222 includes web page administration tools 224, analysis results views 226, and auto-fix wizard 228. The auto-fix wizard 228, which is described in more detail below, leads to manual-fix instructions 230 and modified web page 232.

[0026] Using the web page administration 224, the user specifies that one or more web pages should be automatically analyzed for search engine optimization best practices. A specified web page is added to a list of web pages stored in the web page library 204. At the same time, the web page is submitted to the analysis scheduler 208 to be added to the analysis queue 210 for submission to the automated analysis framework 212. The addition of a web page to the web page library 204 is one of many “analysis events”, as described in more detail below.

[0027] The analysis library 202 represents an extensible framework for adding one or more best practices (also called “analyzes”) into the automated analysis framework 212. Developers can add new analyses to the analysis library 202 and these analyses will be added to the list of best practices against which each web page will be evaluated. In one embodiment, the analysis library 202 is defined in source code (e.g., Java, etc.). In one embodiment, the method and system of the present invention will automatically analyze web pages, and recognize patterns that lead to improved search engine rankings and create and store the resulting analyses in the analysis library 202. These analyses are added to the list of best practices against which each web page is evaluated.

[0028] The analysis scheduler 208 retrieves all the analyses (also known as “best practices”, as indicated above) and the one or more conditions that signify an “analysis event” for each analysis from the analysis library 202. An analysis event can be any one of myriad conditions including, for example, a duration threshold since the web page was last checked, if the web page was modified, if an auto-fix was applied, if the user manually requested that the page be re-analyzed, if the analysis has been changed or updated, etc. In one embodiment, some analysis events may be specific to a single analysis. In another embodiment, some analysis events may apply to multiple or all analyses.

[0029] The analysis scheduler 208 also retrieves the web pages to be analyzed from the web page library 204. The analysis scheduler 208 will monitor or be notified by the database server 118 when analysis events occur for specific web pages. When an analysis event occurs, the analysis scheduler 208 will place the appropriate web pages in the analysis queue 210 for submission to the automated analysis framework 212.

[0030] In one embodiment, the analysis scheduler 208 will conserve computing resources and avoid running one or more analyses if none of the inputs (e.g., the content of the web page, the target keywords, the version of the analysis, etc.) has changed.

[0031] The automated analysis framework 212 analyzes web pages for conformance with SEO best practices. The automated analysis framework 212 reads and runs analyses from the analysis library 202. Analyses include, for example, that the page size is within a certain threshold, the page content includes the primary SEO keywords that the user is optimizing this page for, the page content includes the secondary SEO keywords that the user is optimizing this page for, and the number of inbound links is above a threshold, amongst others. Other analyses include, for example, primary keyword in meta description tag of the web page, secondary keywords in meta key word tag of the web page, existence of keywords in content, registration with major search engines, the age of the web page is within a threshold age, the size of the web page is within a threshold size, the absence of broken links, existence of a meta description tag, and a unique meta description tag within the web site. Other analyses that test the URL, test the structure of the web page, test the content of the web page, and test the links pointing to the page are also included. It will be appreciated that many other analyses not expressly identified herein are within the scope of the present invention.

[0032] The automated analysis framework 212 may retrieve a cached version of the web page being analyzed in evaluating the analysis from the cache manager 218. The cache manager 218 retrieves the cached version of the web page from the web page cache 216. In one embodiment, the web page cache 216 is used to improve performance and minimize the number of web hits to a web page. If the automated analysis framework 212 requests a cached version of a web page from the cache manager 218 and the cached version does not exist in the web page cache 216, or if the cached
In one embodiment, the cache manager 218 may retrieve a newer version of the web page from the internet 214 that was retrieved from the web server 112 and may have been retrieved from a content management server 220, store it in the web page cache 216 and store information regarding the date and time the version was retrieved from the internet 214 in the web page cache 216.

In one embodiment, the user can manually copy the web page or web pages onto a web server or content management server 220. In another embodiment, the system and method of the present invention automatically copy the web page or web pages onto the web server or content management system 220.

In one embodiment, the cache manager 218 monitors the content management server 118 for changes to the web pages or any of the elements of a web page amongst those in the web page library 204. In another embodiment, the cache manager 218 will monitor the internet for changes to the web pages or any elements of a web page amongst those defined in the web page library 204. In one embodiment, when the cache manager 218 detects a change in a web page or elements of a web page amongst those web pages defined in the website library 204, it will store this information in the database server 206 which will then be read by the analysis scheduler 208 and the web page will be submitted to the analysis queue 210 to rerun the analysis for the affected web page.

Once each analysis is run complete, the automated analysis framework 212 then sends the analysis results to the analysis results screens 226, as discussed in more detail below. For each analysis run, the analysis results screens 226, the user can select the analysis for processing by the auto-fix wizard 228, as discussed in more detail below.

FIG. 3 illustrates a functional block diagram and method 300 for the automated analysis framework 212. The method 300 includes the analysis library 302, retrieve analysis block 304, gather all input data block 306, additional analysis inputs 308, web page cache 310, content management server 312, internet 314, evaluate analysis block 316, results database 318, severity criteria 320, run regression block 322, and determine severity block 324. In one embodiment, analysis library 302 is the same as the analysis library 202. In one embodiment, web page cache 310 is the same as the web page cache 216. In one embodiment, the content management server 312 is the same as the content management server 220. In one embodiment, the internet 314 is the same as the internet 214.

The retrieve analysis block 304 reads the details of each analysis from the analysis library 302. The details of each analysis include, for example, the description, the list of additional data required, the test used to check conformance, etc. This information is passed to the gather all input data block 306.

The gather all input data block 306 may retrieve additional inputs from the additional analysis inputs 308 (e.g., the target primary and secondary SEO keywords, etc.), a copy of the web page from the web page cache 310, and/or a copy of the components of the web page from the content management server 312.

The gather all input data block 306 also may retrieve additional information from the internet 314. Such additional information may include, for example, the natural search engine ranking the web page being analyzed receives from each search engine for each target keyword, the number of inbound links to the web page being analyzed, and the anchor text from each of the inbound links, amongst others.

The input data from the gather all input data block 306 is then passed to the evaluate analysis block 316 that executes the analysis results for all the input data. In one embodiment, the evaluate analysis block 316 returns a numeric score for each analysis. The score may be a Boolean 0—fail and 1—pass, or the score may represent volume or scale (e.g., the number of hits, the number of external links, etc.).

The results from, and some of, the input data from the evaluate analysis block 316 are stored in the results database 318.

The system and method of the present invention may use two techniques to indicate the severity of analysis. First, a mathematical technique called “regression analysis” is used to correlate results for each analysis with their corresponding importance in effecting search engine rankings. Second, a rules system with administrator defined rules is used to indicate the severity based on the results of each analysis. In one embodiment, other techniques can be used.

For the “regression analysis”, the run regression block 322 runs a mathematical regression amongst all or some of the results correlating the results of each analysis to the search ranking and in turn establishes ranges for the results that correspond to each severity level. In one embodiment, the search ranking is the organic search ranking that indicates the position of the web page in the portion of search results that do not include paid-for advertising. The run regression block 322 may segment results by industry, time period, and other dimensions. Based on the results of the mathematical regression, the run regression block 322 will calculate severity criteria and store the criteria for each analysis in the severity criteria database 320. In one embodiment, the severity criteria can be stored according to industry, time period, and other dimensions. In other embodiments, different mathematical algorithms may be used.

For the “rules system”, an administrator can define rules to indicate the severity criteria which will also be stored in the severity criteria database 320.

The evaluate analysis block 316 also passes the results to the determine severity block 324 that reads the severity criteria from the severity criteria database 320 and assigns a severity to each analysis.

A first user interface 400, as an exemplary analysis results screen 226, is shown in FIG. 4. The first user interface 400 includes a summary header 402 and a detailed description 404. The summary header 402 includes an “Analysis” heading 406, a results summary 408, and a hide details link 410. The “Analysis” heading 406 is an indication to the user that the first user interface 400 relates to analyses. The results summary 408 is a summary of the results of the analyses run. In one embodiment, the results summary 408 can include information such as, for example, the number of failed analyses and the number of warnings, as described in more detail below. The hide details link 410 allows the user to display only the summary header 402 and not display detailed information about the analysis as shown in the detailed description 404.

The detailed description 404 includes a text description 412, an analysis listing 414, an analysis severity indicators 416, date fields 418, and response options 420. The text description 412 provides a detailed textual account of the results of analyses run. In one embodiment, the text descrip-
tion 412 includes information such as, for example, the percentage of analyses that failed, the percentage of analyses requiring attention, and the percentage of analyses that passed. The analysis status indicators 416 are provided for each analysis in the analysis listing 414. Each analysis status indicator 416 indicates the status or severity of each analysis.

[0048] In one embodiment, the status of an analysis (not shown) may be, for example, “In Progress”, “Complete”, “Inactive”, or “Error”. In one embodiment, the severity of each analysis as indicated by the analysis severity indicators 416 may be, for example, “High”, “Medium”, and “Low”. In another embodiment, the severity of each analysis may be color or otherwise coded. Such coding can include, for example, “Red” (fail), “Yellow” (warning), “Green” (pass), and “Gray” (another indication such as “In Progress”, “Error”, etc.). In one embodiment, the system and method of the present invention may indicate the total number of analyses and the severity or status percentage of analyses. For example, if a total of ten analyses are run with the result that six analyses are Red, then in one embodiment of the present invention, the analysis status indicator 416, or elsewhere on the first user interface 400, would display “Red 60%” as an indication of status and/or severity. Of course, many other indications of status and severity percentages are possible in accordance with the present invention.

[0049] The analysis listing 414 lists analyses that have run or are schedule to be run. In one embodiment, the following analyses are run and thus displayed on the first user interface 400: primary keyword in meta description tag, secondary keywords in meta keywords tag, existence of keywords in content, registered with all major search engines, age of site, page size, and no broken links. In one embodiment, some but not all of the aforementioned analyses can be run. In one embodiment, other analyses can be run in place of or in addition to the aforementioned analyses.

[0050] The date fields 418 indicates the date when each analysis was run. The response options 420 provide selectable options for the user to choose in response to the results of run analyses. In one embodiment, the response options 420 include “Auto-Fix”, “Ignore”, and “More”. In one embodiment, other response options can be included. For certain analyses for each web page, the user can select an “Auto-Fix” button to allow the system 100 in accordance with the present invention to automatically address and remedy a web page that has either failed the analysis or requires attention. For each analysis for each web page, the user can select an “Ignore” button which will display the analysis run with a severity of, for example, “Green” or “Low” regardless of the true severity. In one embodiment, the user can select a “Reset” button (not shown) to revert the analysis back to display its true severity. As described below, for each analysis for each web page, the user can select a “More” button to select other options in response to the running of an analysis, depending on the particular design of the first user interface 400.

[0051] A second user interface 500 of the analysis results screen 226, typically accessed by selecting the “More” button in the first user interface 400, is shown in FIG. 5. The second user interface 500, which can be alternative or in addition to the first user interface 400, includes a web page field 502, a URL field 504, an analysis field 506, a description field 508, a recommendation 510, and a history 512. The web page field 502 identifies the title of the web page against which the analysis was run. The URL field 504 includes the URL for the web page against which the analysis was run. The analysis field 506 identifies the particular analysis to be run. The description field 508 provides an explanatory discussion regarding the operation of search engines and related importance of the analysis for search engine optimization. In one embodiment, the description field 508 of each analysis is user-friendly and targeted for users without a technical or internet marketing background. The recommendation 510 is a detailed discussion regarding how to fix the web page based upon the results of the analysis. In one embodiment, the recommendation 510 can also provide myriad measures that the user can selectively or collectively implement to fix the web page.

[0052] The history 512 displays a list documenting the results of past analysis runs for the particular analysis for the particular web page. The history 512 may be condensed to only show analysis runs in which the severity changed from the severity in the previous analysis run.

[0053] FIG. 6 illustrates a functional block diagram and method 600 for the auto-fix wizard block 228. The method 600 includes the display current content and manual-fix instructions block 602, auto-fix library 606, edit and evaluate content block 608, analysis queue block 610, automated analysis framework block 612, analysis library 614, analysis results 616, generate modified auto-fixed web page block 618, cache manager 620, and content management server 622. In one embodiment, the cache manager 620 is the same as the cache manager 218. In one embodiment, the analysis library 614 is the same as the analysis library 202. In one embodiment, the automated analysis framework 612 is the same as the automated analysis framework 212. In one embodiment, the analysis queue 610 is the same as the analysis queue 210. In one embodiment, the content management server 622 is the same as the content management server 220.

[0054] As discussed above, the automated analysis framework 212 analyzes web pages for conformance with SEO best practices. The automated analysis framework 212 displays the analysis results in the analysis results screen 226.

[0055] Then, for analyses having associated auto-fixes that are defined, the user can click on the auto-fix button in the response options 420, which will read the steps to resolve the problem from the auto-fix library 606 and display a customized list of manual-fix instructions (i.e., a recommendation to improve the condition) specific to a single analysis and a single web page in the current content and manual-fix instructions block 602 to allow the user to manually make the changes to improve the analysis results.

[0056] The user can use a wizard-like interface which guides the user through a series of questions/answers (e.g., changing elements of the web page) and other steps to model a version of the web page that complies with the best practice associated with the analysis and reduces the severity of any issue identified.

[0057] Then, the edit and evaluate content block 608 allows the user to optionally preview the impact of any changes. The analysis is submitted to the analysis queue 610 which will then be processed by the automated analysis framework 612 that will run all analyses that will be impacted by the change. For each analysis, the analysis library 614 includes a list of analyses potentially affected by the change, and generates analysis results 616. In the display current content and manual-fix instructions block 602, the new severity of the current analysis is displayed, as well as the new severities of all analyses potentially affected by the change to ensure that
no analyses change to an unsatisfactory severity. The user may repeat this process as many times as necessary until satisfied with the results.

In the generate modified auto-fixed web page block 618, the user has the option of having the system generate a version of the web page based on the changes specified in the edit and evaluate content block 608 and implementing the web page into the web server 112 or content management server 622. In one embodiment, the generate modified auto-fixed web-page block 618 reads the composite version of the web page from the cache manager 620. In another embodiment, the generate modified auto-fixed web-page block 618 reads the components of a web page from the content management server 622 or a file system.

An exemplary screen 700 of the display current content and manual-fix instructions block 602 is shown in FIG. 7. The exemplary screen 700 of the display current content and manual-fix instructions block 602 includes a web page field 702, a URL field 704, an auto-fix name field 706, a recommendation 708, data entry box(es) 710, and an analyses section 712. The analyses section 712 includes analysis severity indicators 714 and an analysis listing 716.

The web page field 702 identifies the title of the web page against which the analysis was performed. The URL field 704 includes the URL for the web page against which the analysis was run. The auto-fix name field 706 identifies the particular auto-fix that is being run. In the exemplary screen 700, a “primary keyword in meta description tag” is shown as the exemplary auto-fix that is being run in accordance with the present invention. In one embodiment, another auto-fix or other auto-fixes can be run. The recommendation 708 provides an explanatory and targeted discussion regarding the importance of the auto-fix, and provides specific instructions and recommendations to modify the web page to conform the web page to best practices and to resolve any pertinent issues identified.

In this example, the “primary keyword in meta description tag” auto-fix coincidentally shares the same name as the “primary keyword in meta description tag” analysis in the analysis field 506. It will be appreciated that auto-fixes and analyses typically will have different names.

The data entry box 710 allows the user to enter information in response to the information provided in the recommendation 708. The information entered by the user is intended to make the web page conform to the best practices. In the exemplary screen 710, the user is prompted to enter information to constitute a desired meta description tag. In one embodiment, the user is prompted to enter other information relevant to another auto-fix or other auto-fixes that have been run. In one embodiment, a data entry box 710 can be used for a particular web page for a particular analysis. In one embodiment, a plurality of data entry boxes can be used for analysis of a web page. In another embodiment, the exemplary screen 700 does not include any data entry box.

The analysis severity indicators 714 are provided for each analysis in the analysis listing 716. Each analysis severity indicator 716 indicates the severity of each analysis, as described above. The exemplary screen 700 indicates that the primary analysis run is the “primary keyword in meta description tag” and provides a severity indicator for that analysis. In addition, the display current content and manual-fix instructions block 602 displays the severity indicator for other analyses (e.g., “existence of a meta description tag” and “unique meta description tag within site”). In this way, the display current content and manual-fix instructions block 602 can convey the full impact of a web page modification on other analyses even if they are not the analysis of primary or initial interest.

In one embodiment, the display current content and manual-fix instructions block 602 displays a description of the analysis. In one embodiment, the display current content and manual-fix instructions block 602 provides instructions on how to resolve any issues raised by the analysis. In one embodiment, the display current content and manual-fix instructions block 602 provides a wizard-like interface to allow the user to resolve a problem identified by the analysis. In one embodiment, the display current content and manual-fix instructions block 602 provides manual-fix instructions 230 on how to resolve any issues and a preview capability to validate whether any proposed fixes will resolve the problem, and will generate a modified web page 232 with the issues resolved.

In one embodiment, the techniques described above in accordance with the present invention can be applied to optimize landing pages of web sites. In one embodiment, other web pages can be optimized in accordance with the techniques described herein.

FIG. 8 illustrates a block diagram of one embodiment of a system 800 of the present invention. The system 800 includes a first module 810, a second module 820, a third module 830, an Nth module 840, a video display 850, and an input device 860 coupled together through a bus 870. As illustrated, the system 800 includes the modules 810, 820, 830, 840 in one embodiment. In one embodiment, any number of modules can be implemented.

In one embodiment, the routines, steps, and functional blocks executed to implement the embodiments of the disclosure and all of the aforementioned features of the present invention may be variously implemented as computer modules 800, 820, 830, 840. In one embodiment, these computer modules can be a sequence of instructions referred to as “computer programs.”

In one embodiment, user input is provided to one or more of the modules using an input device 860. The input device 860 may be a keyboard, cursor control device, or voice recognition system, for example. In another embodiment, more than one input device may be used. In one embodiment, module output is displayed using a video display 850.

FIG. 9 shows a diagrammatic representation of a machine in an exemplary form of a computer system 900 within which a set of instructions, for causing the machine to perform any one or more of the methodologies discussed herein, may be executed. In alternative embodiments, the machine operates as a standalone device or may be connected (e.g., networked) to other machines. In a networked deployment, the machine may operate in the capacity of a server or a client machine in a client-server network environment, or as a peer machine in a peer-to-peer (or distributed) network environment. In one embodiment, the machine communicates with the server to facilitate operations of the server and/or to access the operations of the server.

The computer system 900 includes a processor 902 (e.g., a central processing unit (CPU), a graphics processing unit (GPU), or both), a main memory 904 and a nonvolatile memory 906, which communicate with each other via a bus 908. In some embodiments, the computer system 900 may be a laptop computer, personal digital assistant (PDA) or mobile phone, for example. The computer system 900 may further
include a video display unit 910 (e.g., a liquid crystal display (LCD) or a cathode ray tube (CRT)). The computer system 900 also includes an alphanumeric input device 912 (e.g., a keyboard), a cursor control device 914 (e.g., a mouse), a disk drive unit 916, a signal generation device 918 (e.g., a speaker) and a network interface device 920. The disk drive unit 916 includes a machine-readable medium (or computer readable medium) 922 on which is stored one or more sets of instructions (e.g., software 924) embodying any one or more of the methodologies or functions described herein. The software 924 may also reside, completely or at least partially, within the main memory 904 and/or within the processor 902 during execution thereof by the computer system 900, the main memory 904 and the processor 902 also constituting machine-readable media. The software 924 may further be transmitted or received over a network 940 via the network interface device 920.

While the machine-readable medium (computer readable medium) 922 is shown in an exemplary embodiment to be a single medium, the term “machine-readable medium” or “computer readable medium” should be taken to include a single medium or multiple media (e.g., a centralized or distributed database, and/or associated caches and servers).

An embodiment of the invention relates to a computer storage product with a computer-readable or machine-accessible medium having executable instructions or computer code thereon for performing various computer-implemented operations. The term “computer-readable medium” or “machine-accessible medium” is used herein to include any medium that is capable of storing or encoding a sequence of executable instructions or computer code for performing the operations described herein. The media and computer code can be those specially designed and constructed for the purposes of the invention, or can be of the kind well known and available to those having ordinary skill in the computer software arts.

Examples of computer-readable media include computer-readable storage media such as: magnetic media such as hard disks, floppy disks, and magnetic tape; optical media such as Compact Disc-Read Only Memories (“CD-ROMs”), DVDs, and holographic devices; magneto-optical media such as optical disks; and hardware devices that are specially configured to store and execute program code, such as Application-Specific Integrated Circuits (“ASICs”), Programmable Logic Devices (“PLDs”), Read Only Memory (“ROM”) devices, and Random Access Memory (“RAM”) devices. Examples of computer code include machine code, such as produced by a compiler, and files containing higher level code that are executed by a computer using an interpreter. For example, an embodiment of the invention may be implemented using Java, C++, or other programming language and development tools. Additional examples of computer code include encrypted code and compressed code. Another embodiment of the invention can be implemented in hard wired circuitry in place of, or in combination with, computer code.

In general, the routines, steps, and functional blocks executed to implement the embodiments of the disclosure and aforementioned features of the present invention may be implemented as part of an operating system or a specific application, component, program, object, module, or sequence of instructions referred to as “computer programs.” The computer programs typically comprise one or more instructions set at various times in various memory and storage devices in a computer, and that, when read and executed by one or more processors in a computer, cause the computer to perform operations to execute elements involving the various aspects of the disclosure.

While the invention has been described with reference to the specific embodiments thereof, it should be understood by those skilled in the art that various changes may be made and equivalents may be substituted without departing from the spirit and scope of the invention as defined by the appended claims. In addition, many modifications may be made to adapt a particular situation, material, composition of matter, method, operation or operations, to the objective, spirit, and scope of the invention. All such modifications are intended to be within the scope of the claims appended hereto. In particular, while the methods disclosed herein have been described with reference to particular operations performed in a particular order, it will be understood that these operations may be combined, subdivided, or reordered to form an equivalent method without departing from the teachings of the invention. Accordingly, unless specifically indicated herein, the order and grouping of the operations is not a limitation of the invention.

We claim:

1. A computer implemented method for search engine optimization (SEO) comprising:
   - receiving, via a computing device, identification of a webpage;
   - adding the webpage to a web page library;
   - upon occurrence of an analysis event, providing the webpage to an automated analysis framework; and
   - running analyses representing SEO best practices against the webpage, each of the analyses associated with input data.

2. The method of claim 1 wherein the analysis event includes at least one of a duration threshold since the webpage was last analyzed, a modification of the webpage, a request to analyze the webpage, a modification of an analysis, and an update to the analysis.

3. The method of claim 2 further comprising, before the adding the webpage to an analysis queue, scheduling the webpage to be added to the analysis queue based on the analysis event.

4. The method of claim 3 further comprising, before the scheduling the webpage to be added to the analysis queue based on the analysis event, determining if the input data associated with an analysis has changed, the input data including at least one of content of the webpage, targeted keywords, and a version of the analysis.

5. The method of claim 1 further comprising providing an analysis library containing a plurality of analyses, the analysis library extensible to receive additional analyses.

6. The method of claim 1 wherein each of the analyses includes at least one of the webpage is within a threshold size, content of the webpage includes a primary SEO keyword for which the webpage is being optimized, content of the webpage includes a secondary SEO keyword for which the webpage is being optimized, and a number of inbound links associated with the webpage is above a threshold.

7. The method of claim 1 wherein each of the analyses includes at least one of presence of a primary keyword in a meta description tag of the webpage, presence of a secondary keyword in a meta description tag of the webpage, existence of keywords in content of the webpage, web page registered
with major search engines, the age of the web page is within a threshold, the size of the web page is within a threshold, and absence of broken links.

8. The method of claim 1 wherein the running analyses representing SEO best practices against the web page further comprises retrieving additional input data including at least one of a targeted primary SEO keyword, one or more secondary SEO keywords, a copy of the web page, and a copy of components of the web page.

9. The method of claim 8 wherein the running analyses representing SEO best practices against the web page further comprises:

executing the analysis based upon the input data;
returning a numeric score for the analysis; and
storing the numeric score in a database.

10. The method of claim 9 wherein the running analyses representing SEO best practices against the web page further comprises determining a severity for the analysis run.

11. The method of claim 10 wherein the running analyses representing SEO best practices against the web page further comprises determining severity criteria by at least one of a regression analysis and rules defined by an administrator.

12. The method of claim 1 further comprising providing response options in a display of results after the running analyses representing SEO best practices against the web page.

13. The method of claim 1 further comprising providing a recommendation regarding how to fix the web page after the running analyses representing SEO best practices against the web page.

14. The method of claim 1 further comprising, in response to selection of one of the response options, automatically fixing the web page.

15. The method of claim 1 further comprising providing an auto-fix wizard to fix the web page after the running analyses representing SEO best practices against the web page.

16. The method of claim 15 further comprising reading detail of an auto-fix from an auto-fix library, the detail of an auto-fix including at least one of a description, a list of additional data required, and a test to check conformance with the analysis.

17. The method of claim 16 further comprising prompting a user to provide information according to the auto-fix.

18. The method of claim 15 further comprising:
providing an auto-fix for a first analysis; and
displaying a severity indication for a second analysis in response to the auto-fix for the first analysis.

19. A machine-readable medium having stored thereon a set of instructions, which when executed by a machine, perform a method comprising:
receiving identification of a web page;
adding the web page to a web page library;
on occurrence of an analysis event, providing the web page to an automated analysis framework; and
running analyses representing SEO best practices against the web page, each of the analyses associated with input data.

20. A computer system comprising:
receiving identification of a web page;
adding the web page to a web page library;
on occurrence of an analysis event, providing the web page to an automated analysis framework; and
running analyses representing SEO best practices against the web page, each of the analyses associated with input data.

* * * * *