US 20170075915A1

(19) United States
(12) Patent Application Publication
LIGHTNER et al.
(10) Pub. No.: US 2017/0075915 A1
(43) Pub. Date: Mar. 16, 2017

(54) SEARCH SUGGESTIONS USING FUZZY-SCORE MATCHING AND ENTITY CO-OCCURRENCE

(71) Applicant: **QBASE, LLC**, Reston, VA (US)

(72) Inventors: **Scott LIGHTNER**, Leesburg, VA (US); **Franz WECKESSER**, Spring Valley, OH (US); **Rakesh DAVE**, Dayton, OH (US); **Sanjay BODDHU**, Dayton, OH (US); **Joseph BECKNELL**, Kettering, OH (US)

(21) Appl. No.: **15/362,685**

(22) Filed: **Nov. 28, 2016**

Related U.S. Application Data

(63) Continuation of application No. 14/950,874, filed on Nov. 24, 2015, now Pat. No. 9,507,834, which is a continuation of application No. 14/558,036, filed on Dec. 2, 2014, now Pat. No. 9,208,204.

(60) Provisional application No. 61/910,900, filed on Dec. 2, 2013.

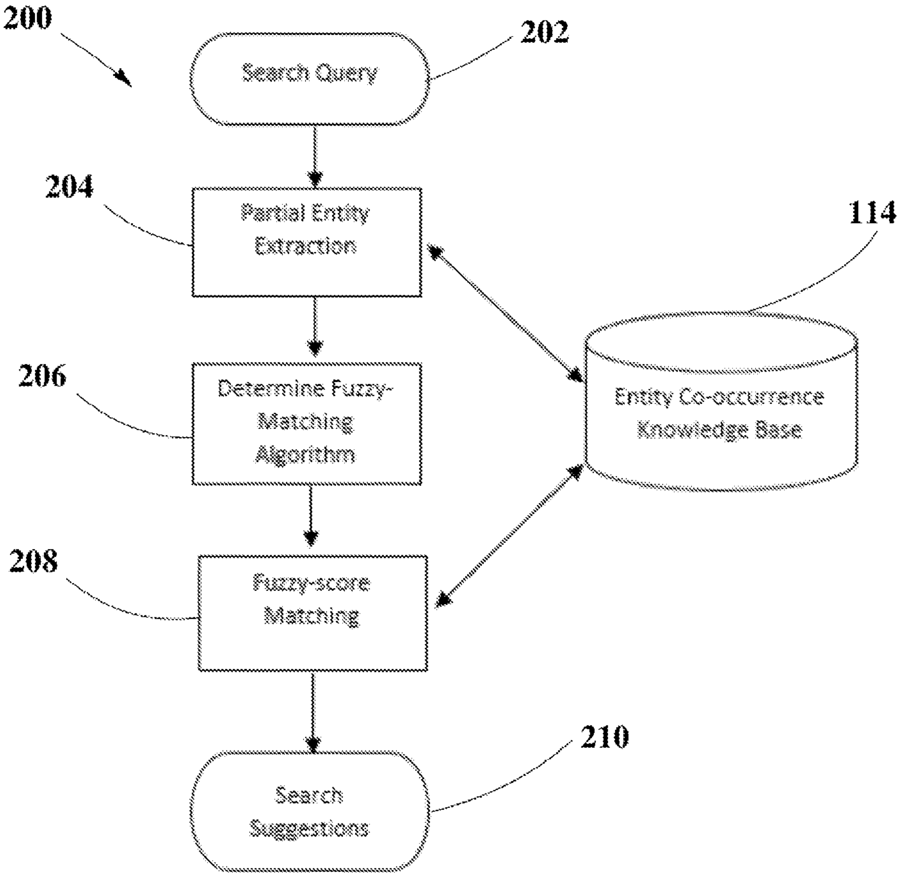
Publication Classification

(51) **Int. Cl.**
G06F 17/30 (2006.01)

(52) **U.S. Cl.**
CPC ***G06F 17/3097*** (2013.01); ***G06F 17/30542*** (2013.01); ***G06F 17/30457*** (2013.01); ***G06F 17/3053*** (2013.01); ***G06F 17/30554*** (2013.01); *G06F 17/30864* (2013.01)

(57) ABSTRACT

A method for generating search suggestions by using fuzzy-score matching and entity co-occurrence in a knowledge base is disclosed. Embodiments of the method may be employed in any search system that may include an entity extraction computer module that may perform partial entity extractions from provided search queries, a fuzzy-score matching computer module that may generate algorithms based on the type of entity extracted and perform a search against an entity co-occurrence knowledge base. The entity co-occurrence knowledge base, which may include a repository where entities may be indexed as entities to entities, entities to topics, or entities to facts among others, may return fast and accurate suggestions to the user to complete the search query. The suggestions may include alternates to the partial query provided by the user that may enhance and save time when performing searches.


200
202
Search Query
204
Partial Entity Extraction
114
206
Determine Fuzzy-Matching Algorithm
Entity Co-occurrence Knowledge Base
208
Fuzzy-score Matching
210
Search Suggestions

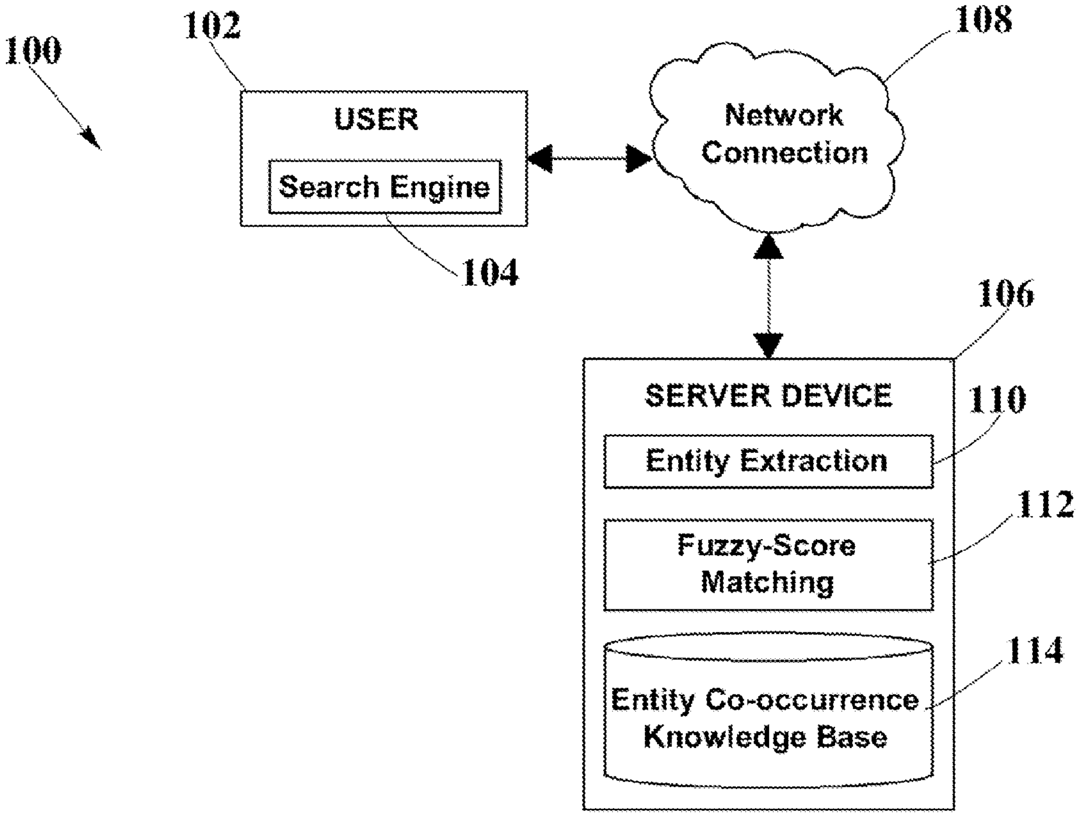

100
102
USER
Search Engine
104
108
Network Connection
106
SERVER DEVICE
110
Entity Extraction
112
Fuzzy-Score Matching
114
Entity Co-occurrence Knowledge Base

FIG. 1

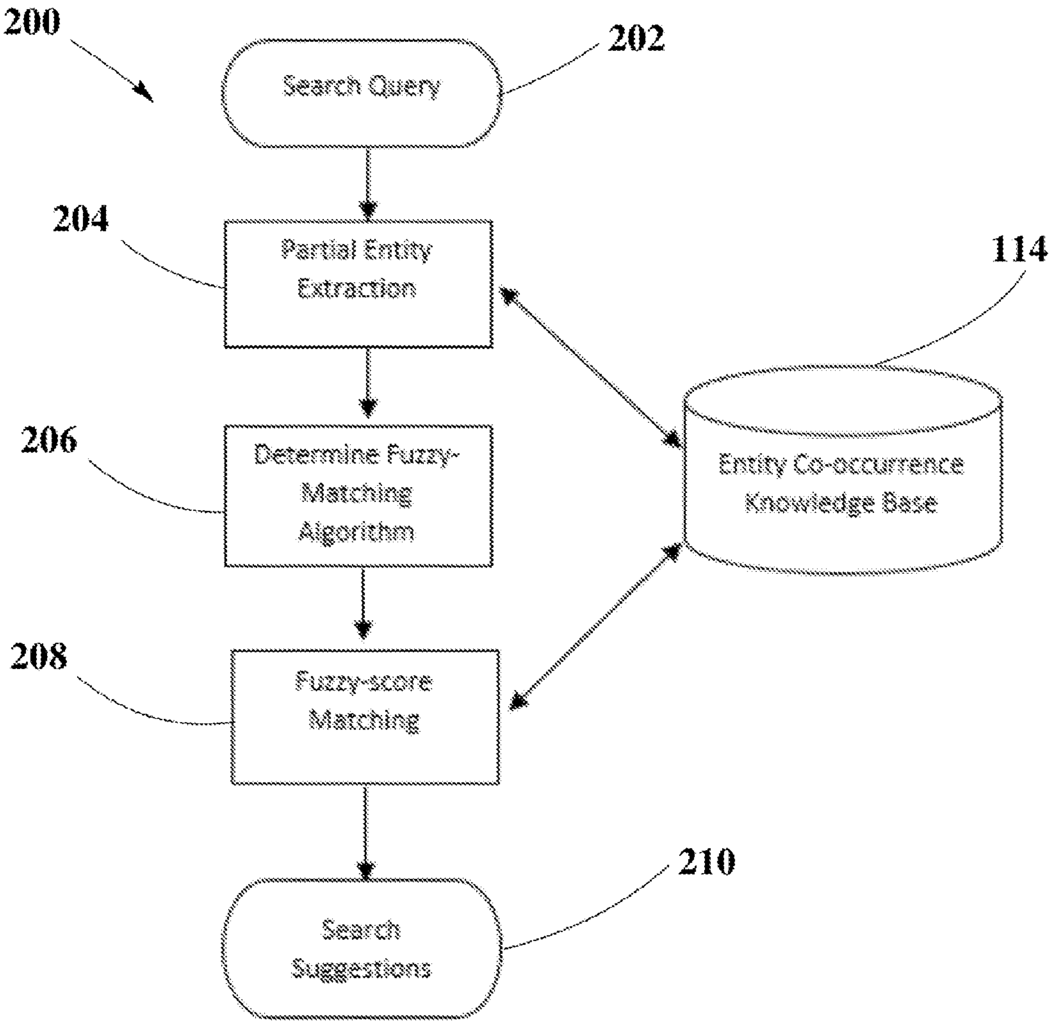

200
202
Search Query
204
Partial Entity Extraction
114
206
Determine Fuzzy-Matching Algorithm
Entity Co-occurrence Knowledge Base
208
Fuzzy-score Matching
210
Search Suggestions

FIG. 2

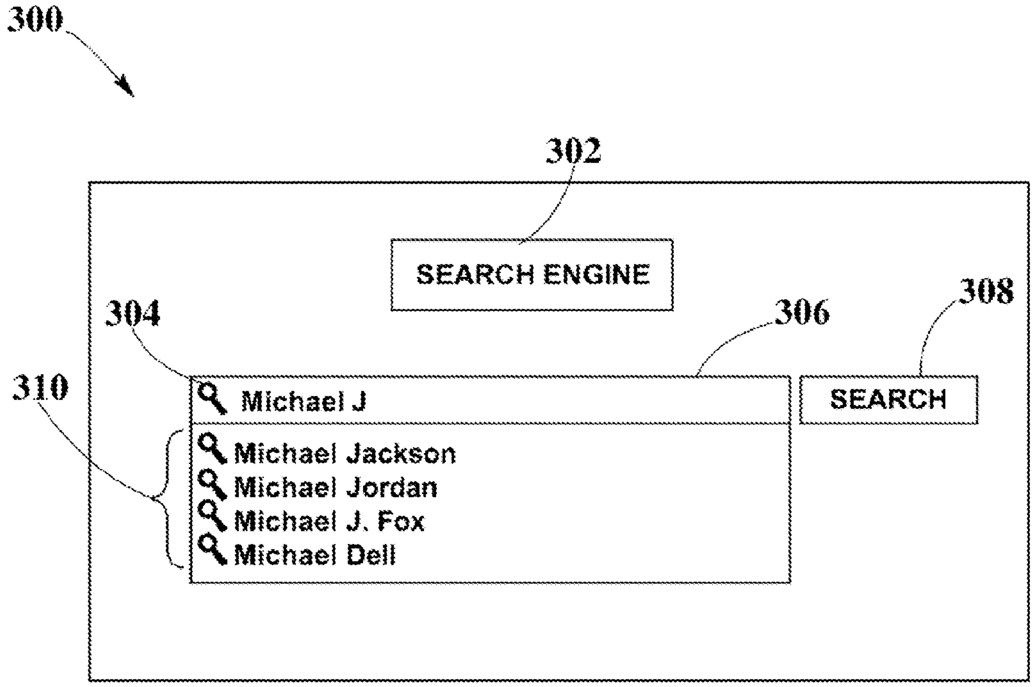

300
302
SEARCH ENGINE
304
306
308
310
Michael J
SEARCH
Michael Jackson
Michael Jordan
Michael J. Fox
Michael Dell

FIG. 3

SEARCH SUGGESTIONS USING FUZZY-SCORE MATCHING AND ENTITY CO-OCCURRENCE

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application is a continuation of U.S. patent application Ser. No. 14/950,874, entitled “Search Suggestions Using Fuzzy-Score Matching and Entity Co-Occurrence,” filed Nov. 24, 2015, issuing as U.S. Pat. No. 9,507,834, which is a continuation of U.S. patent application Ser. No. 14/558,036, entitled “Search Suggestions Using Fuzzy-Score Matching and Entity Co-Occurrence,” filed Dec. 2, 2014, issued as U.S. Pat. No. 9,208,204, which claims the benefit of U.S. Provisional Application No. 61/910,900, filed Dec. 2, 2013, entitled “Search Suggestions Using Fuzzy-Score Matching and Entity Co-Occurrence.” Each above-referenced patent application or patent is incorporated by reference in its entirety.

[0002] This application is related to U.S. patent application Ser. No. 14/557,794, entitled “Method for Disambiguating Features in Unstructured Text,” filed Dec. 2, 2014. This application is related to U.S. patent application Ser. No. 14/558,300, entitled “Event Detection Through Text Analysis Using Trained Event Template Models,” filed Dec. 2, 2014. This application is further related to U.S. patent application Ser. No. 14/558,101, entitled “Non-Exclusionary Search Within In-Memory Databases,” filed Dec. 2, 2014. Each above-referenced patent application or patent is incorporated by reference in its entirety.

TECHNICAL FIELD

[0003] The present disclosure relates generally to query enhancement, and more specifically to search suggestions using fuzzy-score matching and entity co-occurrence in a knowledge base.

BACKGROUND

[0004] Users frequently use search engines for locating information of interest either on the Internet or any database system. Search engines commonly operate by receiving a search query from a user and returning search results to the user. Search results are usually ordered by search engines based on the relevance of each returned search result to the search query. Therefore, the quality of the search query may be significantly important for the quality of search results. However, search queries from users, in most cases, may be written incomplete or partial (e.g., the search query may not include enough words to generate a focused set of relevant results and instead generates a large number of irrelevant results), and sometimes misspelled (e.g., Bill Smith may be incorrectly spelled as Bill Smitth).

[0005] One common approach to improve the quality of the search results is to enhance the search query. One way to enhance the search queries may be by generating possible suggestions based on the user’s input. For this, some approaches propose methods for identifying candidate query refinements for a given query from past queries submitted by one or more users. However, these approaches are based on query logs that sometimes may lead the user to results that may not be of interest. There are other approaches using different techniques that may not be accurate enough. Thus, there still exists a need for methods that improve or enhance search queries from users to get more accurate results.

SUMMARY

[0006] A method for generating search suggestions by using fuzzy-score matching and entity co-occurrence in a knowledge base is disclosed. In one aspect of the present disclosure, the method may be employed in a search system that may include a client/server type architecture. In one embodiment, the search system may include a user interface to a search engine in communication with one or more server devices over a network connection. The server device may include an entity extraction computer module, a fuzzy-score matching computer module, and an entity co-occurrence knowledge base database. The knowledge base may be built as an in-memory database and may also include other hardware and/or software components such as one or more search controllers, multiple search nodes, collections of compressed data, and a disambiguation computer module. One search controller may be selectively associated with one or more search nodes. Each search node may be capable of independently performing a fuzzy key search through a collection of compressed data and returning a set of scored results to its associated search controller.

[0007] In another aspect of the present disclosure, the method may include an entity extraction module that may perform partial entity extractions from provided search queries to identify whether the search query refers to an entity, and if so, to what type of entity it refers. Furthermore, the method may include a fuzzy-score matching module that may spawn algorithms based on the type of entity extracted and perform a search against an entity co-occurrence knowledge base. Additionally, the query text parts that are not detected as corresponding to entities are treated as conceptual features, such as topics, facts, and key phrases, that can be employed for searching the entity co-occurrence knowledge base. In an embodiment, the entity co-occurrence knowledge base includes a repository where entities may be indexed as entities to entities, entities to topics, or entities to facts among others, which facilitates the return of fast and accurate suggestions to the user to complete the search query.

[0008] In one embodiment, a method is disclosed. The method comprises receiving, by an entity extraction computer, user input of search query parameters from a user interface, extracting, by the entity extraction computer, one or more entities from the search query parameters by comparing the search query parameters with an entity co-occurrence database having instances of co-occurrence of the one or more entities in an electronic data corpus and identifying at least one entity type corresponding to the one or more entities in the search query parameters, and selecting, by a fuzzy-score matching computer, a fuzzy matching algorithm for searching the entity co-occurrence database to identify one or more records associated with the search query parameters, wherein the fuzzy matching algorithm corresponds to the at least one identified entity type. The method further includes searching, by the fuzzy-score matching computer, the entity co-occurrence database using the selected fuzzy matching algorithm and forming one or more suggested search query parameters from the one or more records based on the search, and presenting, by the fuzzy-score matching computer, the one or more suggested search query parameters via the user interface.

[0009] In another embodiment, a system is provided. The system includes one or more server computers having one or more processors executing computer readable instructions for a plurality of computer modules including an entity extraction module configured to receive user input of search query parameters from a user interface, the entity extraction module being further configured to extract one or more entities from the search query parameters by comparing the search query parameters with an entity co-occurrence database having instances of co-occurrence of the one or more entities in an electronic data corpus and identifying at least one entity type corresponding to the one or more entities in the search query parameters. The system further includes a fuzzy-score matching module configured to select a fuzzy matching algorithm for searching the entity co-occurrence database to identify one or more records associated with the search query parameters, wherein the fuzzy matching algorithm corresponds to the at least one identified entity type. The fuzzy-score matching module being further configured to search the entity co-occurrence database using the selected fuzzy matching algorithm and form one or more suggested search query parameters from the one or more records based on the search, and present the one or more suggested search query parameters via the user interface.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] The present disclosure can be better understood by referring to the following figures. The components in the figures are not necessarily to scale, emphasis instead being placed upon illustrating the principles of the disclosure. In the figures, reference numerals designate corresponding parts throughout the different views.

[0011] FIG. **1** is a block diagram illustrating an exemplary system environment in which one embodiment of the present disclosure may operate;

[0012] FIG. **2** is a flowchart illustrating a method for search suggestions using fuzzy-score matching and entity co-occurrence in a knowledge base, according to an embodiment; and

[0013] FIG. **3** is a diagram illustrating an example of a user interface through which a search suggestion may be produced using fuzzy matching and entity co-occurrence in a knowledge base of FIGS. **1-2**.

DETAILED DESCRIPTION

[0014] The present disclosure is here described in detail with reference to embodiments illustrated in the drawings, which form a part hereof. Other embodiments may be used and/or other changes may be made without departing from the spirit or scope of the present disclosure. The illustrative embodiments described in the detailed description are not meant to be limiting of the subject matter presented herein.

[0015] As used herein, the following terms may have the following definitions:

[0016] "Entity extraction" refers to computer executed information processing methods for extracting electronic information including, but not limited to, names, places, and organizations.

[0017] "Corpus" refers to a collection of one or more documents and other electronic data.

[0018] "Features" is any information which is at least partially derived from an electronic document.

[0019] "Module" refers to a computer hardware and/or software components suitable for carrying out at least one or more tasks.

[0020] "Fact" refers to objective relationships between features.

[0021] "Entity knowledge base" refers to a computer database containing features and entities.

[0022] "Query" refers to an electronic request to retrieve information from one or more suitable databases.

[0023] "Topic" refers to a set of thematic information which is at least partially derived from a corpus.

[0024] Reference will now be made to the exemplary embodiments illustrated in the drawings, and specific language will be used here to describe the same. It will nevertheless be understood that no limitation of the scope of the invention is thereby intended. Alterations and further modifications of the inventive features illustrated here, and additional applications of the principles of the inventions as illustrated here, which would occur to one skilled in the relevant art and having possession of this disclosure, are to be considered within the scope of the invention.

[0025] Embodiments of the present disclosure introduce a novel search suggestion generation mechanism, which is different from the existing mechanisms that are based on mining and ranking the activity of the search system's global users past search queries. In an embodiment, the presented search suggestions mechanism is based on employing an entity co-occurrence knowledge base in its core, along with fuzzy matching computer module and an entity extraction computer module. The entity co-occurrence knowledge base is a repository, such as an electronic database, where entities may be indexed as entities to entities, entities to topics, or entities to facts among others, and stored so as to allow faster and weighted responses. The user's partial, as well as complete, queries can be processed on-the-fly to detect entities (e.g., via entity extraction), misspelled variations (e.g., via fuzzy matching) of the entities and other conceptual features and these features are employed to search (e.g., via fuzzy score matching) entity co-occurrence knowledge base to suggest search queries, as well as possible expansions/suggestions of the accurate entities intended by the user. This leads to a more accurate and improved search experience, as discussed in further detail in FIGS. **1-3** below.

[0026] FIG. **1** is a block diagram of a search computer system **100** in accordance with the present disclosure. The search system **100** may include one or more user interfaces **102** to a search engine **104** in communication with a server device **106** over a network **108**. In this embodiment, the search system **100** may be implemented in one or more special purpose computers and computer modules referenced below, including via a client/server type architecture. However, the search system **100** may be implemented using other computer architectures (for example, a stand-alone computer, a mainframe system with terminals, an ASP model, a peer to peer model and the like). In an embodiment, the search computer system **100** includes a plurality of networks such as, a local area network, a wide area network, the internet, a wireless network, a mobile phone network and the like.

[0027] A search engine **104** may include a user interface, such as a web-based tool that enables users to locate information on the World Wide Web. Search engine **104** may also include user interface tools that enable users to locate information within internal database systems. Server device

106, which may be implemented in a single server device **106** or in a distributed architecture across a plurality of server computers, may include an entity extraction module **110**, a fuzzy-score matching module **112**, and an entity co-occurrence knowledge base database **114**.

[0028] Entity extraction module **110** may be a hardware and/or software module configured to extract and disambiguate on-the-fly independent entities from a given set of queries such as a query string, partial query, structured data and the like. Examples of entities may include people, organizations, geographic locations, dates and/or time. During the extraction, one or more feature recognition and extraction algorithms may be employed. Also, a score may be assigned to each extracted feature, indicating the level of certainty of the feature being correctly extracted with the correct attributes. Taking into account the feature attributes, the relative weight or relevance of each of the features may be determined. Additionally, the relevance of the association between features may be determined using a weighted scoring model.

[0029] Fuzzy-score matching module **112** may include a plurality of algorithms that may be selected according to the type of entity being extracted from a given search query. The function of the algorithms may be to determine whether the given search query received via user input and other searched strings identified by the algorithm are similar to each other, or approximately match a given pattern string. Fuzzy matching may also be known as fuzzy string matching, inexact matching, and approximate matching. Entity extraction module **110** and fuzzy-score matching module **112** may work in conjunction with entity co-occurrence knowledge base **114** to generate search suggestions for the user.

[0030] According to various embodiments, entity co-occurrence knowledge base **114** may be built, but is not limited to, as an in-memory database and may include components, such as one or more search controllers, multiple search nodes, collections of compressed data, and a disambiguation module. One search controller may be selectively associated with one or more search nodes. Each search node may be capable of independently performing a fuzzy key search through a collection of compressed data and returning a set of scored results to its associated search controller.

[0031] Entity co-occurrence knowledge base **114** may include related entities based on features and ranked by a confidence score. Various methods for linking the features may be employed, which may essentially use a weighted model for determining which entity types are most important, which have more weight, and, based on confidence scores, determine how confident the extraction of the correct features has been performed.

[0032] FIG. **2** is a flowchart illustrating a method **200** for generating search suggestions using fuzzy-score matching and entity co-occurrence in a knowledge base. Method **200** may be implemented in a search system **100**, similar to that described by FIG. **1**.

[0033] In one embodiment, method **200** may initiate when a user starts typing a search query in step **202** into a search engine interface **102**, as described in FIG. **1**. As the search query is typed in step **202**, search system **100** may perform an on-the-fly process. According to various embodiments, search query input in step **202** may be either complete or partial, either correctly spelled or misspelled. Followed, at search system **100**, a partial entity extraction step **204** from the search query input of step **202** may be performed. The partial entity extraction step **204** may run a quick search against entity co-occurrence knowledge base **114** to identify whether the search query that was input in step **202** is an entity, and if so, what type of entity it is. According to various embodiments, search query input of step **202** may then refer to a person, an organization, the location of a place, and a date among others. Once the entity type of the search query input is identified, fuzzy-score matching module **112** may select a corresponding fuzzy matching algorithm, step **206**. For example, if search query was identified as an entity that is referring to a person, then fuzzy-score matching module **112** may select the string matching algorithm for persons, for example, such as by extracting different components of the person's name including first, middle, last, and title. In another embodiment, if search query was identified as an entity that is referring to an organization, then fuzzy-score matching module **112** may select the string matching algorithm for organizations, which can include identifying terms like school, university, corp, inc, and the like. Fuzzy-score matching module **112** may then select the string matching algorithm that corresponds to the type of identified entity in the search query input so as to excel the search. Once the string matching algorithm is adjusted to the type of identified entity, a fuzzy-score matching step **208** may be performed.

[0034] In fuzzy-score matching step **208**, extracted entity or entities, as well as non-entities, may be searched and compared against entity co-occurrence knowledge base **114**. Extracted entity or entities may include incomplete names of persons, for example first name and the first character of the last name, abbreviations of organizations, for example "UN" that may stand for "United Nations", short forms, and nicknames among others. Entity co-occurrence knowledge base **114** may already have registered a plurality of records indexed as an structured data, such as entity to entity, entity to topics, and entity to facts, among others. The latter may allow fuzzy-score matching in step **208** to happen in a very fast way. Fuzzy-score matching in step **208** may use, but is not limited to, a common string metric such as Levenshtein distance, strcmp95, ITF scoring, and the like. Levenshtein distance between two words may refer to the minimum number of single-character edits required to change one word into the other.

[0035] Finally, once fuzzy-score matching step **208** finishes comparing and searching search query against all records in the entity co-occurrence knowledge base **114**, the record that dominates the most or is the closest to match the given pattern string (i.e., the search query input of step **202**) may be selected as first candidate for a search suggestion in step **210**. Other records with less proximity to match the given pattern string may be placed under the first candidate in a descending order. Search suggestion in step **210** may then be presented to the user in a drop down list of possible matches that the user may or may not ignore.

[0036] FIG. **3** is an example user interface **300** in accordance with the method for generating search suggestions using fuzzy-score matching and entity co-occurrence in a knowledge base, as discussed in FIGS. **1**-**2** above. In this example, a user through a search engine interface **302**, similar to that described by FIG. **1**, inputs a partial query **304** in a search box **306**. By a way of illustration and not by way of limitation, partial query **304** may be an incomplete name of a person such as "Michael J", as shown in FIG. **3**. It may be considered a partial query **304** because the user may not have yet selected search button **308**, or otherwise submitted the partial query **304** to search system **100** to perform an actual search and obtain results.

[0037] Following the method **200** (FIG. **2**), as the user types "Michael J", the entity extraction module **110** performs a quick search on-the-fly of the first word (Michael) against entity co-occurrence knowledge base **114** to identify the type of entity, in this example, the entity may refer to the name of a person. Consequently, fuzzy-score matching module **112** may select a string match algorithm tailored for names of persons. Name of persons may be found in databases written in different forms such as using only initials (short forms), or first name and first character of the last name, or first name, initial of the middle name and last name, or any combination thereof. Fuzzy-score matching module **112** may use a common string metric such as Levenshtein distance to determine and assign a score to the entity, topic, or fact within entity co-occurrence knowledge base **114** that may match the entity "Michael". In this example, Michael matches with a great amount of records having that name. However, as the user types the following character "J", fuzzy-score matching module **112** may perform another comparison based on Levenshtein distance against all co-occurrences with Michael with the entity co-occurrence knowledge base **114**. Entity co-occurrence knowledge base **114** may then select all possible matches with the highest scores for "Michael J". For example, fuzzy-score matching module **112** may return search suggestions **310** such as "Michael Jackson", "Michael Jordan", "Michael J. Fox", or even "Michael Dell" in some cases to the user. The user may then be able to select from the drop down list one of the persons suggested as to complete the search query. Expanding on the aforementioned example, a query like "Michael the basketball player", would lead to the suggestion of "Michael Jordan", based on the results returned by searching entity co-occurrence knowledge base for "Michael" in person entity name variations and "the basketball player" in the co-occurrence features like key phrases, facts, and topics. Another example can be "Alexander the actor", would lead to the suggestion of "Alexander Polinsky". Those skilled in the art will realize that the presently existing search platforms cannot generate suggestions in the aforementioned manner.

[0038] While various aspects and embodiments have been disclosed, other aspects and embodiments are contemplated. The various aspects and embodiments disclosed are for purposes of illustration and are not intended to be limiting, with the true scope and spirit being indicated by the following claims.

[0039] The foregoing method descriptions and the process flow diagrams are provided merely as illustrative examples and are not intended to require or imply that the steps of the various embodiments must be performed in the order presented. As will be appreciated by one of skill in the art the steps in the foregoing embodiments may be performed in any order. Words such as "then," "next," etc. are not intended to limit the order of the steps; these words are simply used to guide the reader through the description of the methods. Although process flow diagrams may describe the operations as a sequential process, many of the operations can be performed in parallel or concurrently. In addition, the order of the operations may be re-arranged. A process may correspond to a method, a function, a procedure, a subroutine, a subprogram, etc. When a process corresponds to a function, its termination may correspond to a return of the function to the calling function or the main function.

[0040] The various illustrative logical blocks, modules, circuits, and algorithm steps described in connection with the embodiments disclosed here may be implemented as electronic hardware, computer software, or combinations of both. To clearly illustrate this interchangeability of hardware and software, various illustrative components, blocks, modules, circuits, and steps have been described above generally in terms of their functionality. Whether such functionality is implemented as hardware or software depends upon the particular application and design constraints imposed on the overall system. Skilled artisans may implement the described functionality in varying ways for each particular application, but such implementation decisions should not be interpreted as causing a departure from the scope of the present invention.

[0041] Embodiments implemented in computer software may be implemented in software, firmware, middleware, microcode, hardware description languages, or any combination thereof. A code segment or machine-executable instructions may represent a procedure, a function, a subprogram, a program, a routine, a subroutine, a module, a software package, a class, or any combination of instructions, data structures, or program statements. A code segment may be coupled to another code segment or a hardware circuit by passing and/or receiving information, data, arguments, parameters, or memory contents. Information, arguments, parameters, data, etc. may be passed, forwarded, or transmitted via any suitable means including memory sharing, message passing, token passing, network transmission, etc.

[0042] The actual software code or specialized control hardware used to implement these systems and methods is not limiting of the invention. Thus, the operation and behavior of the systems and methods were described without reference to the specific software code being understood that software and control hardware can be designed to implement the systems and methods based on the description here.

[0043] When implemented in software, the functions may be stored as one or more instructions or code on a non-transitory computer-readable or processor-readable storage medium. The steps of a method or algorithm disclosed here may be embodied in a processor-executable software module which may reside on a computer-readable or processor-readable storage medium. A non-transitory computer-readable or processor-readable media includes both computer storage media and tangible storage media that facilitate transfer of a computer program from one place to another. A non-transitory processor-readable storage media may be any available media that may be accessed by a computer. By way of example, and not limitation, such non-transitory processor-readable media may comprise RAM, ROM, EEPROM, CD-ROM or other optical disk storage, magnetic disk storage or other magnetic storage devices, or any other tangible storage medium that may be used to store desired program code in the form of instructions or data structures and that may be accessed by a computer or processor. Disk and disc, as used here, include compact disc (CD), laser disc, optical disc, digital versatile disc (DVD), floppy disk, and Blu-ray disc where disks usually reproduce data magnetically, while discs reproduce data optically with lasers. Combinations of the above should also be included within the scope of computer-readable media. Additionally, the operations of a method or algorithm may reside as one or any combination or set of codes and/or instructions on a non-transitory processor-readable medium and/or computer-readable medium, which may be incorporated into a computer program product.

[0044] The preceding description of the disclosed embodiments is provided to enable any person skilled in the art to make or use the present invention. Various modifications to these embodiments will be readily apparent to those skilled in the art, and the generic principles defined here may be applied to other embodiments without departing from the spirit or scope of the invention. Thus, the present invention is not intended to be limited to the embodiments shown here but is to be accorded the widest scope consistent with the following claims and the principles and novel features disclosed here.

What is claimed is:

1. A method comprising:

in response to comparing, by a server, a search query against a plurality of records of an in-memory database, identifying, by the server, an entity corresponding to the search query and a type of the entity corresponding to the search query, wherein the search query is received from a client, wherein the records contain a co-occurrence knowledge dataset; and

in response to selecting, by the server, a fuzzy-score matching algorithm based on the type, determining, by the server, that the entity is sufficiently similar to a record of the records based on the fuzzy-score matching algorithm and presenting, by the server, a content of the record to the client, wherein the fuzzy-score matching algorithm uses a metric having a minimum number of single-character edits that are needed to change a first word in the search query into a second word in the record.

2. The method of claim **1**, wherein the entity corresponds to at least one of a person content, an organization content, a location of a place content, or a date content.

3. The method of claim **1**, wherein the in-memory database is indexed as structured data.

4. The method of claim **3**, wherein the structured data includes at least one of entity-to-entity index, an entity-to-topic index, or an entity-to-fact index.

5. The method of claim **1**, further comprising:

selecting, by the server, the record as a first candidate to be presented on the client as a search suggestion.

6. The method of claim **5**, further comprising:

presenting, by the server, at least some of the records other than the record to the client, positionally under the record in the search suggestion, in a descending order.

7. The method of claim **1**, wherein the metric employs a Levenshtein distance.

8. The method of claim **1**, wherein the presenting of the content of the record is before the search query is finalized at the client.

9. The method of claim **1**, wherein the in-memory database includes a search controller, a search node, and a collection of compressed data.

10. The method of claim **1**, wherein the comparing, the identifying, the selecting, and the determining are in real-time as the search query is being received from the client.

11. A system comprising:

a server programmed to:

responsive to comparing, by the server, a search query against a plurality of records of an in-memory database, identify, by the server, an entity corresponding to the search query and a type of the entity corresponding to the search query, wherein the search query is received from a client, wherein the records contain a co-occurrence knowledge dataset; and

responsive to selecting, by the server, a fuzzy-score matching algorithm based on the type of the entity, determine, by the server, that the entity is sufficiently similar to a record of the records based on the fuzzy-score matching algorithm and present, by the server, a content of the record to the client, wherein the fuzzy-score matching algorithm uses a metric having a minimum number of single-character edits that are needed to change a first word in the search query into a second word in the record.

12. The server of claim **11**, wherein the entity corresponds to at least one of a person content, an organization content, a location of a place content, or a date content.

13. The server of claim **11**, wherein the in-memory database is indexed as structured data.

14. The server of claim **13**, wherein the structured data includes at least one of entity-to-entity index, an entity-to-topic index, or an entity-to-fact index.

15. The server of claim **11**, wherein the server is programmed to select the record as a first candidate to be presented on the client as a search suggestion.

16. The server of claim **11**, wherein the server is programmed to present at least some of the records other than the record to the client, positionally under the record in the search suggestion, in a descending order.

17. The server of claim **11**, wherein the metric employs a Levenshtein distance.

18. The server of claim **11**, wherein the server is programmed to present the content of the record before the search query is finalized at the client.

19. The server of claim **11**, where the in-memory database includes a search controller, a search node, and a collection of compressed data.

20. The server of claim **11**, wherein the server is programmed to compare, identify, select and determine in real-time as the search query is being received from the client.

* * * * *