US 20160239500A1

(19) United States

(12) Patent Application Publication

(10) Pub. No.: US 2016/0239500 A1

(43) Pub. Date: Aug. 18, 2016

DAVE et al.

(54) SYSTEM AND METHODS FOR EXTRACTING FACTS FROM UNSTRUCTURED TEXT

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

(72) Inventors: **Rakesh DAVE**, Dayton, OH (US); **Sanjay BODDHU**, Dayton, OH (US)

(21) Appl. No.: **15/136,731**

(22) Filed: **Apr. 22, 2016**

Related U.S. Application Data

(63) Continuation of application No. 14/557,802, filed on Dec. 2, 2014.

(60) Provisional application No. 61/910,880, filed on Dec. 2, 2013.

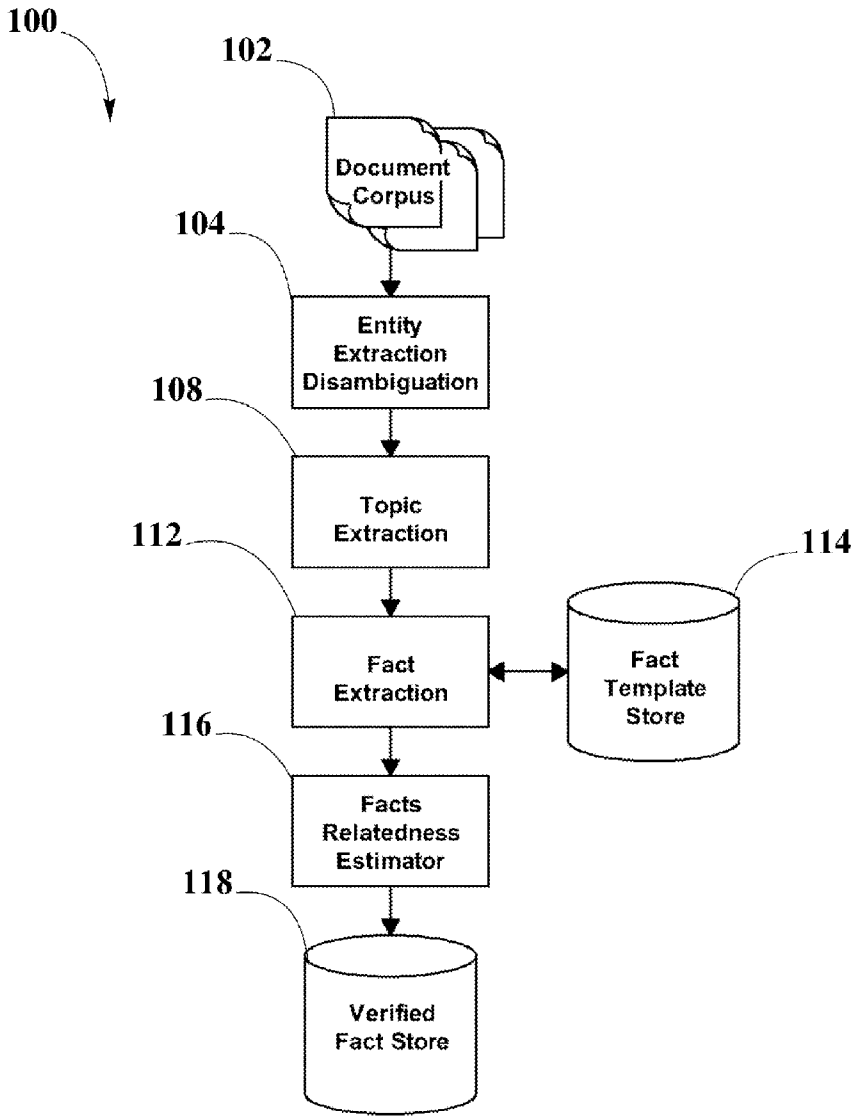
Publication Classification

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

(52) **U.S. Cl.**
CPC *G06F 17/3071* (2013.01); *G06F 17/30675* (2013.01)

(57) **ABSTRACT**

A system and method for extracting facts from unstructured text files are disclosed. Embodiments of the disclosed system and method may receive a text file as input and perform extraction and disambiguation of entities, as well as extract topics and facts. The facts are extracted by comparing against a fact template store and associating facts with events or topics. The extracted facts are stored in a data store.



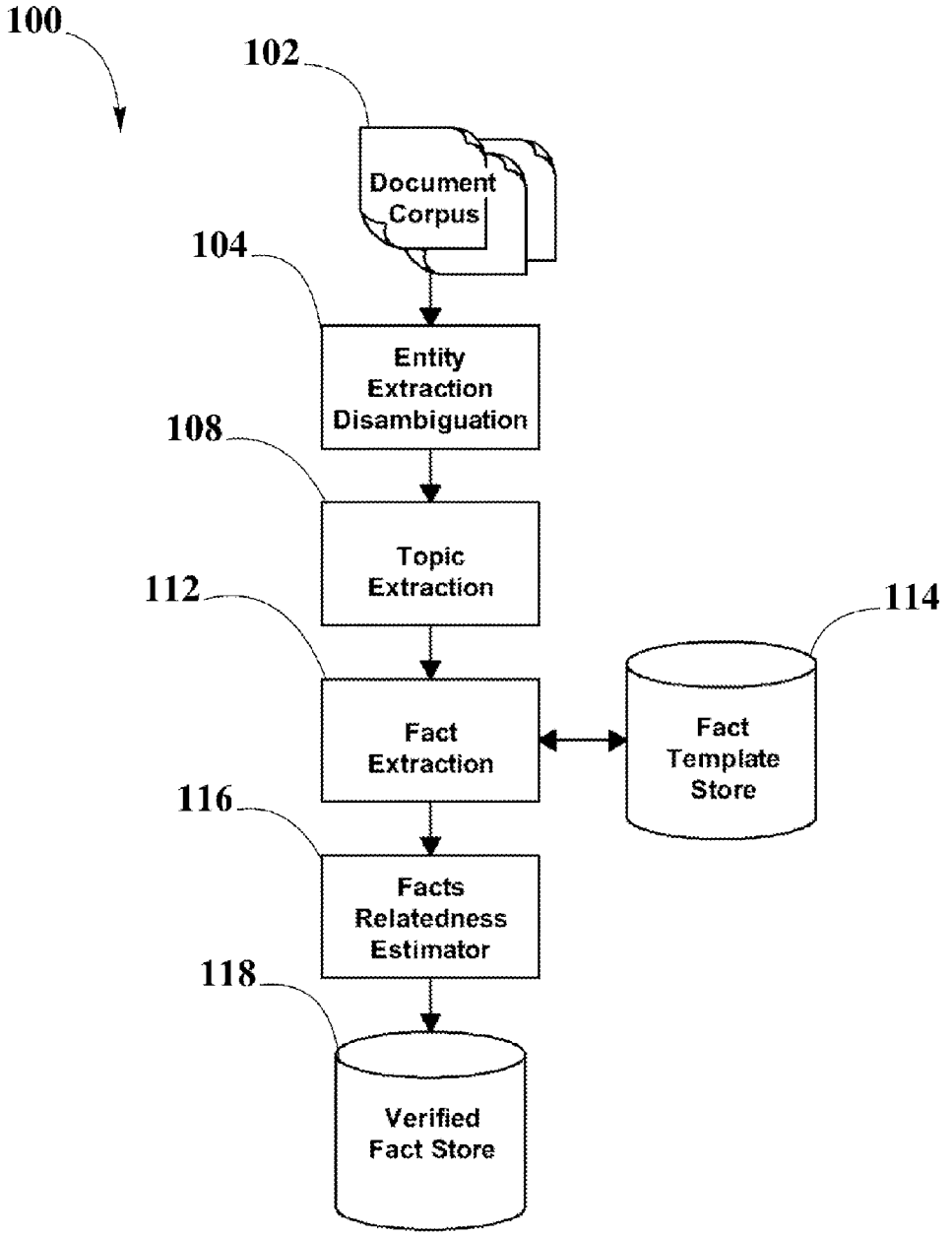

100
102
Document Corpus
104
Entity Extraction Disambiguation
108
Topic Extraction
112
Fact Extraction
114
Fact Template Store
116
Facts Relatedness Estimator
118
Verified Fact Store

FIG. 1

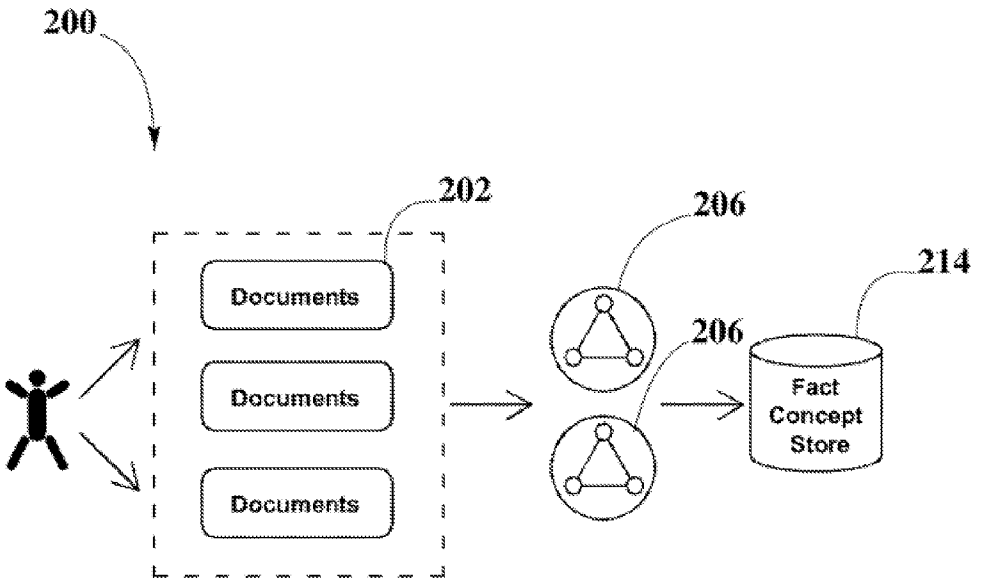

200
202
206
214
206
Documents
Documents
Documents
Fact
Concept
Store

FIG. 2

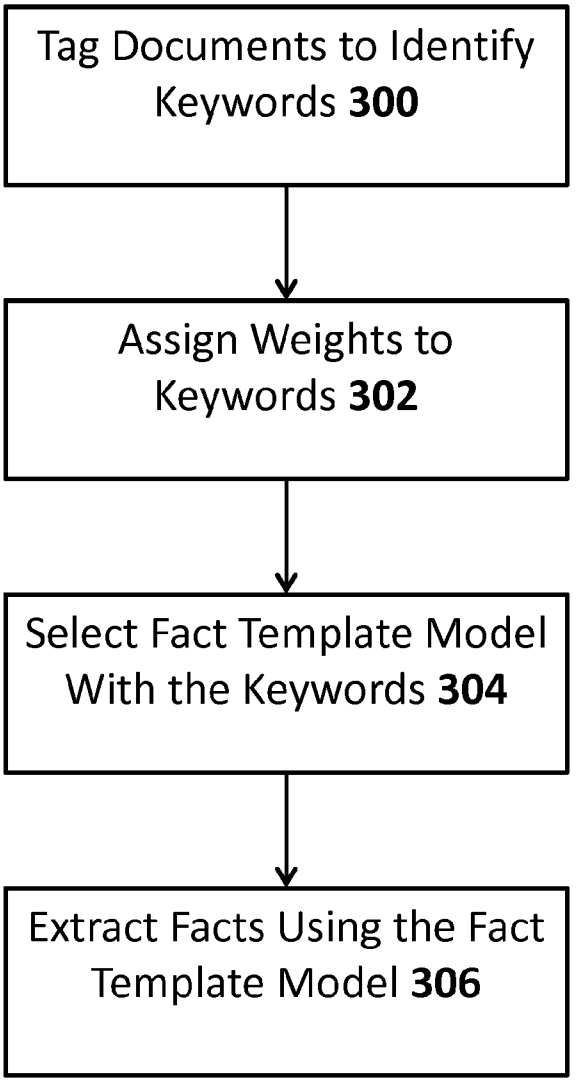

Tag Documents to Identify Keywords 300
Assign Weights to Keywords 302
Select Fact Template Model With the Keywords 304
Extract Facts Using the Fact Template Model 306

FIG. 3

SYSTEM AND METHODS FOR EXTRACTING FACTS FROM UNSTRUCTURED TEXT

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application is a continuation of U.S. Non-Provisional application Ser. No. 14/557,802, entitled "System and Method for Extracting Facts from Unstructured Text," filed on Dec. 2, 2014, which claims the benefit of priority to U.S. Provisional Application No. 61/910,880, filed Dec. 2, 2013, entitled "System and Method for Extracting Facts From Unstructured Text," all of which are fully incorporated by reference herein for all purposes.

[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; U.S. patent application Ser. No. 14/558,300, entitled "Event Detection Through Text Analysis Using Trained Event Template Models," filed Dec. 2, 2014; U.S. patent application Ser. No. 14/558,076, entitled "Method For Automated Discovery Of New Topics," filed Dec. 2, 2014; and U.S. patent application Ser. No. 14/558,342, entitled "Event Detection Through Text Analysis Using Dynamic Self Evolving/Learning Module," filed Dec. 2, 2014; each of which are incorporated herein by reference in their entirety for all purposes.

TECHNICAL FIELD

[0003] The present disclosure relates in general to information data mining from document sources, and more specifically to extraction of facts from documents.

BACKGROUND

[0004] Electronic document corpora may contain vast amounts of information. For a person searching for specific information in a document corpus, identifying key information may be troublesome. Manually crawling each document and highlighting or extracting important information may even be impossible depending on the size of the document corpus. At times a reader may only be interested in facts or asserted information. The use of intelligent computer systems for extracting features in an automated matter may be commonly used to aid in fact extraction. However, current intelligent systems fail to properly extract facts and associate them with other extracted features such as entities, topics, events and other feature types.

[0005] Thus a need exists for a method of extracting facts and accurately associating them with features to improve accuracy of information.

SUMMARY

[0006] A system and method for extracting facts from unstructured text are disclosed. The system includes an entity extraction computer module used to extract and disambiguate independent entities from an electronic document, such as a text file. The system may further include a topic extractor computer module configured to determine a topic related to the text file. The system may extract possible facts described in the text by comparing text string structures against a fact template store. The fact template store may be built by revising documents containing facts and recording a commonly used fact sentence structure. The extracted facts may then be associated with extracted entities and topics to determine a confidence score that may serve as an indication of the accuracy of the fact extraction.

[0007] In one embodiment, a method is disclosed. The method comprises receiving, by an entity extraction computer, an electronic document having unstructured text and extracting, by the entity extraction computer, an entity identifier from the unstructured text in the electronic document. The method further includes extracting, by a topic extraction computer, a topic identifier from the unstructured text in the electronic document, and extracting, by a fact extraction computer, a fact identifier from the unstructured text in the electronic document by comparing text string structures in the unstructured text to a fact template database, the fact template database having stored therein a fact template model identifying keywords pertaining to specific fact identifiers and corresponding keyword weights. The method further includes associating, by a fact relatedness estimator computer, the entity identifier with the topic identifier and the fact identifier to determine a confidence score indicative of a degree of accuracy of extraction of the fact identifier.

[0008] In another embodiment, a system is disclosed. The system comprises one or more server computers having one or more processors executing computer readable instructions for a plurality of computer modules. The computer modules include an entity extraction module configured to receive an electronic document having unstructured text and extract an entity identifier from the unstructured text in the electronic document, a topic extraction module configured to extract a topic identifier from the unstructured text in the electronic document, and a fact extraction module configured to extract a fact identifier from the unstructured text in the electronic document by comparing text string structures in the unstructured text to a fact template database, the fact template database having stored therein a fact template model identifying keywords pertaining to specific fact identifiers and corresponding keyword weights. The system further includes a fact relatedness estimator module configured to associate the entity identifier with the topic identifier and the fact identifier to determine a confidence score indicative of a degree of accuracy of extraction of the fact identifier.

[0009] In yet another embodiment, a non-transitory computer readable medium having stored thereon computer executable instructions. The instructions comprise receiving, by an entity extraction computer, an electronic document having unstructured text, extracting, by the entity extraction computer, an entity identifier from the unstructured text in the electronic document, and extracting, by a topic extraction computer, a topic identifier from the unstructured text in the electronic document. The instructions further include extracting, by a fact extraction computer, a fact identifier from the unstructured text in the electronic document by comparing text string structures in the unstructured text to a fact template database, the fact template database having stored therein a fact template model identifying keywords pertaining to specific fact identifiers and corresponding keyword weights, and associating, by a fact relatedness estimator computer, the entity identifier with the topic identifier and the fact identifier to determine a confidence score indicative of a degree of accuracy of extraction of the fact identifier.

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 diagram of a fact extraction system, according to an embodiment.

[0012] FIG. **2** is diagram of a system for training a fact concept store, according to an embodiment.

[0013] FIG. **3** is a flow chart of a method for building a fact template store of FIG. **2**, according to an embodiment.

DETAILED DESCRIPTION

[0014] The present disclosure is herein 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 information processing methods for extracting information such as names, places, and organizations from electronic documents.

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

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

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

[0020] "Facts" refers to asserted information about features found in an electronic document.

[0021] 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.

[0022] The present disclosure describes a system and method for detecting, extracting and validating facts from a document source.

[0023] Various embodiments of the systems and methods disclosed here collect data from different sources in order to identify independent events. Embodiments of the present disclosure introduce a framework for extracting facts from unstructured text. The embodiments disclosed herein accurately associate extracted facts with other features (like topics, linguistic features, disambiguated entities and disambiguated entity types) retrieved from the text and employ a fact template store containing commonly used fact sentence structures. This approach allows the assignment of confidence scores to extracted facts and leads to significantly improved accuracy. The following embodiments are performed by a central computer server system having one or more processors executing computer readable instructions corresponding to a plurality of special purpose computer modules described in FIGS. **1-3** below.

[0024] FIG. **1** depicts an embodiment of a system **100** for extracting facts from an electronic document. Embodiments of the disclosed system may be implemented in various operating environments that include personal computers, server computers, handheld or laptop devices, multiprocessor systems, microprocessor-based systems, programmable consumer electronics, network PCs, minicomputers, mainframe computers, and distributed computing environments.

[0025] The document corpus computer module **102** may provide an input of an electronic document containing unstructured text such as, for example, a news feed article, a file from a digital library, a blog, a forum, a digital book and/or any file containing natural language text.

[0026] The process may involve crawling through document file received from the corpus **102**. An electronic document may include information in unstructured text format which may be crawled using natural language processing techniques (NLP). Some NLP techniques include, for example, removing stop words, tokenization, stemming and part-of speech tagging among others known in the art.

[0027] An individual file may first go through an entity extraction computer module **104** where entities (e.g., a person, location, or organization name) are identified and extracted. Entity extraction module **104** may also include disambiguation methods which may differentiate ambiguous entities. Disambiguation of entities may be performed in order to attribute a fact to an appropriate entity. A method for entity disambiguation may include, for example, comparing extracted entities and co-occurrences with other entities or features against a knowledge base of co-occurring features in order to identify specific entities the document may be referring to. Other methods for entity disambiguation may also be used and are included within the scope of this disclosure. In an embodiment, entity extraction computer module **104** may be implemented as a hardware and/or software module in a single computer or in a distributed computer architecture.

[0028] The file may then go through a topic extractor computer module **108**. Topic extractor module **108** may extract the theme or topic of a single document file. In most cases a file may include a single topic, however a plurality of topics may also exist in a single document. Topic extraction techniques may include, for example, comparing keywords against models built with a multi-component extension of latent Dirichlet allocation (MC-LDA), among other techniques for topic identification. A topic may then be appended to a fact in order to provide more accurate information.

[0029] System **100** may include a fact extractor computer module **112**. Fact extractor module **112** may be a hardware and/or software computer module executing programmatic logic that may extract facts by crawling through the document. Fact extractor module **112** may compare text structures against fact models stored in a fact template store **114** in order to extract and determine the probability of an extracted fact and the associated fact type.

[0030] In the illustrated embodiment, once all features are extracted, a fact relatedness estimator computer module **116** may correlate all features in order to determine a fact relation to other features and assign a confidence score that may serve as an indication that an extracted fact is accurate. Fact relatedness estimator module **116** may calculate a confidence score based on a text distance between parts of text from where a fact was extracted and where a topic or entity was extracted. For example, consider the fact example "President said the bill will pass" extracted from a document where the identified topic was "immigration". Fact relatedness estimator module **116** may measure the distances between the fact

sentence "President said the bill will pass" and the sentence from where the topic "immigration" was extracted. The shorter the distance in text, the more likelihood that the fact is indeed related to immigration. The fact relatedness estimator module **116** may also calculate confidence score by comparing co-occurring entities in the same document file. For example, considering the same example used before the entity "president" may be mentioned at different parts in the document. A co-occurrence of an entity mentioned in a fact with the same entity in a different part of the document may increase a confidence score associated with the fact. The distances between co-occurring entities in relation to facts may also be used in determining confidences scores. Distances in text may be calculated using methods such as tokenization or any other NLP methods.

[0031] Whenever the confidence score for an extracted fact exceeds a predetermined threshold, such fact may be stored in a verified fact store **118**. Verified fact store **118** may be a computer database used by various applications in order to query for different facts associated with the purpose of a given application.

[0032] Those skilled in the art will realize that FIG. **1** illustrates an exemplary embodiment and is in no way limiting the scope of the invention. Additional modules for extracting different features not illustrated in FIG. **1** may also be included and are to be considered within the scope of the invention. As those of skill in the art will realize, all hardware and software modules described in the present disclosure may be implemented in a single special purpose computer or in a distributed computer architecture across a plurality of special purpose computers.

[0033] FIG. **2** is an embodiment of a training computer system **200** for building a fact template store **214**. A plurality of documents **202** may be tagged, for example by a computer process, in order to identify key words pertaining to specific facts and assign weights to those keywords. For example, an embodiment of a fact template model **206** may be "The President said the bill will pass." The tagging process of the system **200** can identify, tag and record the sentence structure of the fact. In the example, to build a model the person may identify the keyword "said" preceded by an entity (e.g., the "President") and proceeded by some string (e.g., "the bill will pass") which may represent the value of the fact. The model may then be stored in fact template store **214** along with metadata such as for example, a count of how many times that sentence structure is repeated across different documents, a fact type classification, a confidence score that serves as an indication of how strongly the sentence structure may resemble a fact. Fact template models **206** may be used in subsequent text comparisons in order to extract facts from document files.

[0034] FIG. **3** is an embodiment of a method for building a fact template store of FIG. **2**. In step **300**, the computer system **200** (FIG. **2**) tags electronic documents in a corpus of documents to identify keywords pertaining to facts. In step **302**, the system **200** assigns weights to tagged keywords. In step **304**, the system **200** selects a fact template model having the identified keywords (from other electronic documents in the corpus) and stores the fact template in the fact template store database along with the metadata, as discussed above in connection with FIG. **2**. Finally, in step **306**, the fact template model is used in text comparisons in the process of fact extraction, as discussed in FIG. **1** above.

[0035] 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.

[0036] The various illustrative logical blocks, modules, circuits, and algorithm steps described in connection with the embodiments disclosed herein 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.

[0037] 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.

[0038] 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 herein.

[0039] 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 herein 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 herein, 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.

[0040] 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 herein 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 herein but is to be accorded the widest scope consistent with the following claims and the principles and novel features disclosed herein.

What is claimed is:

1. A method comprising:

extracting, by a server, an entity identifier and a topic identifier from a first portion of unstructured text in a text file, the unstructured text comprising a plurality of strings;

comparing, by the server, the strings to data stored in a data structure, the data comprising a fact template model containing a keyword related to a fact identifier and a keyword weight associated with the fact identifier;

extracting, by the server, the fact identifier from a second portion of the unstructured text of the text filed based upon comparing the strings to the data comprising the fact template model;

associating, by the server, the entity identifier with the topic identifier and the fact identifier; and

determining, by the server, a score based on the associating, wherein the score is indicative of a degree of accuracy of the extracting of the fact identifier, wherein the score is based on a spatial distance between the first portion and the second portion.

2. The method of claim **1**, wherein the spatial distance is determined via a token.

3. The method of claim **1**, wherein the text file comprises a plurality of co-occurring entity identifiers, wherein the score is based on comparing the co-occurring entity identifiers.

4. The method of claim **1**, wherein the fact template model includes metadata.

5. The method of claim **4**, wherein the metadata includes a count of a number of times a sentence structure corresponding to the fact template model is repeated across a plurality of electronic documents comprising the text file.

6. The method of claim **4**, wherein the metadata comprises the score.

7. The method of claim **1**, where at least two of the extracting, the comparing, the extracting, the associating, and the determining are distributed among a plurality of computers.

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

determining, by the server, whether the score exceeds a predetermined threshold;

in response to the score exceeding the predetermined threshold, storing, by the server, the fact identifier in a second data structure.

9. The method of claim **1**, wherein the spatial distance is determined via a natural language processing technique.

10. A device comprising:

a processor;

a memory storing a set of instructions executable by the processor to perform a method comprising:

extracting, by the processor, an entity identifier and a topic identifier from a first portion of an unstructured text in a text file, wherein the unstructured text comprises a plurality of strings, wherein the unstructured text comprises a second portion;

comparing, by the processor, the strings to a data stored in a data structure, wherein the data comprises a fact template model which identifies a keyword related to a fact identifier and a keyword weight associated with the fact identifier;

extracting, by the processor, the fact identifier from the second portion based on the comparing;

associating, by the processor, the entity identifier with the topic identifier and the fact identifier;

determining, by the processor, a score based on the associating, wherein the score is indicative of a degree of accuracy of the extracting of the fact identifier, wherein the score is based on a spatial distance between the first portion and the second portion.

11. The device of claim **10**, wherein the spatial distance is determined via a token.

12. The device of claim **1**, wherein the text file comprises a plurality of co-occurring entity identifiers, wherein the score is based on comparing the co-occurring entity identifiers.

13. The device of claim **1**, wherein the fact template model includes metadata.

14. The device of claim **13**, wherein the metadata includes a count of a number of times a sentence structure corresponding to the fact template model is repeated across a plurality of electronic documents comprising the text file.

15. The device of claim **13**, wherein the metadata comprises the score.

16. The device of claim **10**, where at least two of the extracting, the comparing, the extracting, the associating, and the determining are distributed among a plurality of computers.

17. The device of claim **10**, further comprising:

determining, by the processor, whether the score exceeds a predetermined threshold;

in response to the score exceeding the predetermined threshold, storing, by the processor, the fact identifier in a second data structure.

18. The device of claim **10**, wherein the spatial distance is determined via a natural language processing technique.

* * * * *