(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
23 September 2021 (23.09.2021)

WIPO | PCT

(10) International Publication Number
WO 2021/186364 A1

(51) International Patent Classification:
G06F 16/20 (2019.01) *G06F 17/00* (2019.01)
G06F 16/30 (2019.01) *G06F 40/00* (2020.01)

(21) International Application Number:
PCT/IB2021/052233

(22) International Filing Date:
17 March 2021 (17.03.2021)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
202041011410 17 March 2020 (17.03.2020) IN

(71) Applicant: **L&T TECHNOLOGY SERVICES LIMITED** [IN/IN]; DLF IT SEZ Park, Block 3, 2nd Floor, 1/124, Mount Poonamallee Road, Ramapuram, Chennai 600089 (IN).

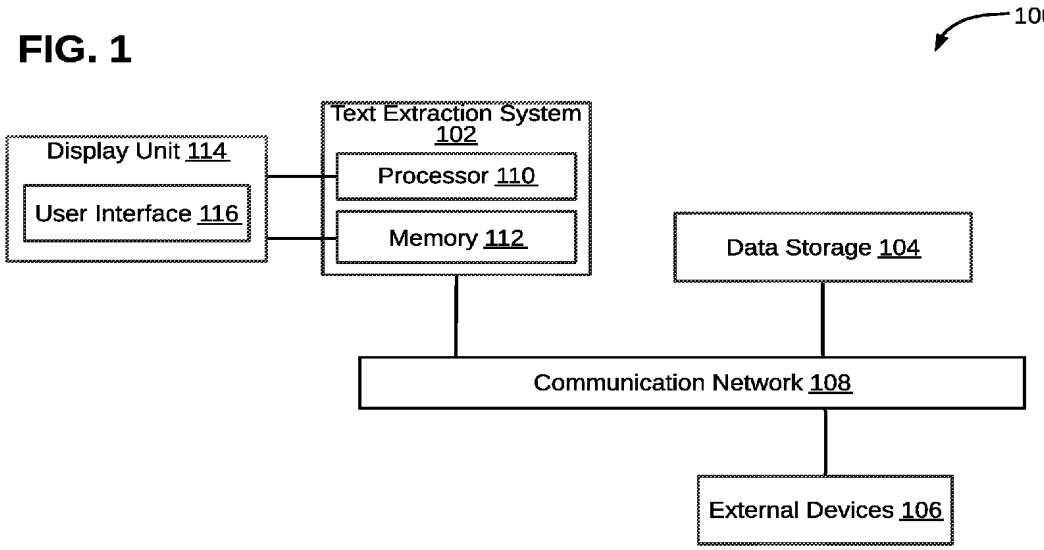
(72) Inventors: **SINGH, Madhusudan**; House No.B-603, Ajmera Stone Park, 1st Cross, Electronic City - 1, Bangalore 560100 (IN). **GHOSH DASTIDAR, Aritra**; Flat no 06, 181 Banamali Banerjee Road, Haridevpur, Landmark: Sodepur, Shani Mandir, Kolkata 700082 (IN). **SHA, Ajay**; House No. 298, c/o Shambhu Sha, Beside Post Office, N.S Road, Jaigaon Dist, Alipurduar 736182 (IN). **VANAPALLI VENKATA RAMESH RAYULU, Nirmal**; House No. 6/74, Road # 4, Deepthisri Nagar, Madinaguda, Hyderabad 500050 (IN). **HALDER, Kaushik**; Flat – 4A; Block – A, Ideal Apartment, 152- Raja Rajendra Lal Mitra Road, Beleghata, Kolkata 700010 (IN).

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

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH,

(54) Title: EXTRACTING TEXT-ENTITIES FROM A DOCUMENT MATCHING A RECEIVED INPUT

FIG. 1



(57) Abstract: This disclosure relates to a method (500) and system (102) for extracting text from a document based on a received input. The method (500) includes deciphering (502) a character pattern associated with received input. The deciphering of character pattern includes identifying (602) a character type associated with each of a plurality of characters in received input, assigning (604) to each of the plurality of characters a character code, creating (606) one or more clusters for received input in response to assigning character code to each of plurality of characters and replacing (608), for each of the one or more clusters, at least one contiguous occurrence of the same character code with a single occurrence of the same character code to generate the character pattern. The method (500) further includes extracting (504), from the document, at least one text-entity matching the received input, based on character pattern deciphered for received input.

WO 2021/186364 A1

[Continued on next page]

GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to the identity of the inventor (Rule 4.17(i))*

Published:

— *with international search report (Art. 21(3))*

— *in black and white; the international application as filed contained color or greyscale and is available for download from PATENTSCOPE*

EXTRACTING TEXT-ENTITIES FROM A DOCUMENT MATCHING A RECEIVED INPUT

DESCRIPTION

TECHNICAL FIELD

[001] This disclosure relates generally to text extraction, and more particularly to a method of extracting, from a document, one or more text-entities matching a received input.

BACKGROUND

[002] Conventionally, for knowledge-gathering approach, data may be extracted from a document by reading the document and such approach may be prevalent well before the advent of the age of computation. However, manually browsing through a huge number of documents to gather data for analysis may lead to errors. Moreover, such an approach may be time consuming and labor-intensive.

[003] Some available techniques for extracting text-entities from documents may use Natural Language Processing (NLP) related to a given text-entity that may correspond to a received input or a user query. However, such techniques are limited to matching only certain attributes of the received input with the text-entities of the document. Consequently, such techniques may fail to extract all the text-entities from the document relevant to the received input. Further, these techniques may not provide for dynamic training of a rule engine for improving the accuracy of extracting text-entities from the document without requiring manual intervention.

[004] Accordingly, there is a need for an improved method of extracting relevant text-entities that match a received input.

SUMMARY

[005] In accordance with an embodiment, a method for extracting text from a document based on a received input is disclosed. The method may include deciphering a character pattern associated with the received input. The received input may include a plurality of characters. In accordance with an embodiment, deciphering the character pattern may include identifying a character type from a plurality of character types associated with each of the plurality of characters in the received input, and assigning, to each of the plurality of characters, a character code from a plurality of character codes based on the identified character type from the plurality of character types. In accordance with an embodiment, deciphering the character pattern may

further include creating one or more clusters for the received input in response to assigning the character code to each of the plurality of characters. Each of the one or more clusters may include at least one contiguous occurrence of the same character code. In accordance with an embodiment, deciphering the character pattern may include replacing, for each of the one or more clusters, the at least one contiguous occurrence of the same character code with a single occurrence of the same character code to generate the character pattern. The method may further include extracting, from the document, at least one text-entity matching the received input, based on the character pattern deciphered for the received input.

[006] In accordance with an embodiment, a system for extracting text from a document based on a received input is disclosed. The system includes a processor and a memory communicatively coupled to the processor. The memory is configured to store processor-executable instructions. The processor-executable instructions, on execution, may cause the processor to decipher a character pattern associated with the received input. The received input may include a plurality of characters. In accordance with an embodiment, deciphering the character pattern by the processor may include identifying a character type from a plurality of character types associated with each of the plurality of characters in the received input, and assigning, to each of the plurality of characters, a character code from a plurality of character codes based on the identified character type from the plurality of character types. In accordance with an embodiment, deciphering the character pattern by the processor may further include creating one or more clusters for the received input in response to assigning the character code to each of the plurality of characters. Each of the one or more clusters may include at least one contiguous occurrence of the same character code. In accordance with an embodiment, deciphering the character pattern by the processor may further include replacing, for each of the one or more clusters, the at least one contiguous occurrence of the same character code with a single occurrence of the same character code to generate the character pattern. The processor instructions may further cause the processor to extract, from the document, at least one text-entity matching the received input, based on the character pattern deciphered for the received input.

[007] It is to be understood that both the foregoing general description and the following detailed description are exemplary and explanatory only and are not restrictive of the invention, as claimed.

BRIEF DESCRIPTION OF THE DRAWINGS

[008] The accompanying drawings, which are incorporated in and constitute a part of this disclosure, illustrate exemplary embodiments and, together with the description, serve to explain the disclosed principles.

[009] FIG. 1 is a block diagram that illustrates an environment for extracting text from a document based on a received input, in accordance with an embodiment of the present disclosure.

[010] FIG. 2 is a functional block diagram of a text extraction system for extracting text from a document based on a received input, in accordance with some other embodiments of the present disclosure.

[011] FIG. 3 is a flowchart that illustrates an exemplary method for identifying a character pattern in a document and generating a set of rules, in accordance with an embodiment of the present disclosure.

[012] FIG. 4 illustrates a model/rule engine of a text extraction system for extracting a character pattern in a document based on a set of rules, in accordance with an embodiment of the present disclosure.

[013] FIG. 5 illustrates a flowchart of an exemplary method of extracting text from a document based on a received input, in accordance with an embodiment of the present disclosure.

[014] FIG. 6 illustrates a flowchart of an exemplary method of deciphering a character pattern associated with the received input, in accordance with an embodiment of the present disclosure.

[015] FIG. 7 illustrates a flowchart of an exemplary method for extracting, from a document, at least one text-entity matching the received input, in accordance with an embodiment of the present disclosure.

[016] FIG. 8 illustrates a flowchart of an exemplary method of deciphering a character pattern associated with each text-entity of the at least one text-entity from the document, in accordance with an embodiment of the present disclosure.

[017] FIG. 9 illustrates a flowchart of an exemplary method for removing at least one noise text-entity from the extracted at least one text-entity matching the received input, in accordance with an embodiment of the present disclosure.

DETAILED DESCRIPTION

[018] Exemplary embodiments are described with reference to the accompanying drawings. Wherever convenient, the same reference numbers are used throughout the drawings to refer

to the same or like parts. While examples and features of disclosed principles are described herein, modifications, adaptations, and other implementations are possible without departing from the spirit and scope of the disclosed embodiments. It is intended that the following detailed description be considered as exemplary only, with the true scope and spirit being indicated by the following claims.

[019] The following described implementations may be found in the disclosed method and system for extracting text from a document based on a received input. The disclosed system (referred as text extraction system) may use a rule engine to extract text entities for improving the accuracy of extracting text-entities from the document without requiring manual intervention. The disclosed text extraction system may facilitate extracting, from a document, text-entities matching an input-token. The disclosed text extraction system may provide for time and labor efficient training of the rule engine for improving the accuracy of extracting text-entities from the document. The disclosed text extraction system may facilitate identification of various different type of relevant text-entities in a document, while minimizing chances of missing any relevant text-entities.

[020] Referring now to FIG. 1, a block diagram illustrates an environment for extracting text from a document based on a received input, in accordance with an embodiment of the present disclosure. The environment 100 includes a text extraction system 102, a data storage 104, external devices 106, and a communication network 108. In accordance with an embodiment, the text extraction system 102 may include a processor 110 and a memory 112. There is further shown a display unit 114 that may include a user interface (UI) 116.

[021] The text extraction system 102 may be communicatively coupled to the data storage 104 and the external devices 106, via the communication network 108. A user (not shown in FIG. 1) may be associated with the text extraction system 102 or the external devices 106.

[022] The text extraction system 102 may include suitable logic, circuitry, interfaces, and/or code that may be configured to receive an input that may include a plurality of characters. The input may correspond to a user query to extract a relevant text entity from a document. In accordance with an embodiment, the input may correspond to an input token that includes a plurality of characters. In accordance with an embodiment, the text extraction system 102 may be configured to decipher the character pattern associated with the received input. In accordance with an embodiment, the text extraction system 102 may be configured to extract, from the document, at least one text-entity matching the received input, based on the character pattern deciphered for the received input.

[023] In accordance with an embodiment, the text extraction system 102 may use an application for application-specific deployment that includes software and/or logic to extract text from the document based on the received input. By way of example, the text extraction system 102 may be implemented as a plurality of distributed cloud-based resources by use of several technologies that are well known to those skilled in the art. Other examples of implementation of the text extraction system 102 may include, but are not limited to, a web/cloud server, an application server, a media server, and a Consumer Electronic (CE) device (such as, but not limited to, a desktop, a laptop, a smartphone and a tablet).

[024] The processor 110 may be communicatively coupled to the memory 112. The processor 110 may include suitable logic, circuitry, interfaces, and/or code that may be configured to extract text from the document based on the input token or received input. The processor 110 may be implemented based on a number of processor technologies, which may be known to one ordinarily skilled in the art. Examples of implementations of the processor 110 may be a Graphics Processing Unit (GPU), a Reduced Instruction Set Computing (RISC) processor, an Application-Specific Integrated Circuit (ASIC) processor, a Complex Instruction Set Computing (CISC) processor, a microcontroller, Artificial Intelligence (AI) accelerator chips, a co-processor, a central processing unit (CPU), and/or a combination thereof. The processor 110 may be communicatively coupled to, and communicates with, the memory 112.

[025] The memory 112 may include suitable logic, circuitry, and/or interfaces that may be configured to store instructions executable by the processor 110. Additionally, the memory 112 may be configured to store program code of one or more software applications that may incorporate the program code of the one or more text extraction algorithms. The memory 112 may be configured to store any received data (such as, a received query as input token from a user) or generated data (such as, clusters of same character codes) associated with storing, maintaining, and executing the text extraction system 102 used to extract text entities from the document based on the received input. Examples of implementation of the memory 112 may include, but are not limited to, Random Access Memory (RAM), Read Only Memory (ROM), Electrically Erasable Programmable Read-Only Memory (EEPROM), Hard Disk Drive (HDD), a Solid-State Drive (SSD), a CPU cache, and/or a Secure Digital (SD) card.

[026] In accordance with an embodiment, the data storage 104 may store documents and input received from users for the text extraction system 102, character patterns associated with text entities in documents, and data associated with the documents for access by users of the text extraction system 102. For example, the data storage 104 may store metadata indicative of

location of character patterns of text entities within documents along with the documents, and may be accessed via the communication network 108.

[027] In accordance with an embodiment, the data storage 104 may store data structures for use in extraction of text entities from documents and input. While the example of FIG. 1 includes a single data storage (the data storage 104) located elsewhere in the environment 100 from the text extraction system 102, in some embodiments, the data storage 104 may also be included as a part of the text extraction system 102.

[028] By way of an example, and not limitation, the data storage 104 may use computer-readable storage media that includes tangible or non-transitory computer-readable storage media including, but not limited to, Compact Disc Read-Only Memory (CD-ROM) or other optical disk storage, magnetic disk storage or other magnetic storage devices (e.g., Hard-Disk Drive (HDD)), flash memory devices (e.g., Solid State Drive (SSD), Secure Digital (SD) card, other solid state memory devices), or any other storage medium which may be used to carry or store particular program code in the form of computer-executable instructions or data structures and which may be accessed by a general-purpose or special-purpose computer. Combinations of the above may also be included within the scope of computer-readable storage media.

[029] The external devices 106 may include suitable logic, circuitry, interfaces, and/or code that may be configured to transmit input token or a user query to the text extraction system 102 for extracting at least one text-entity matching the received input or the user query. The external devices 106 may be capable of communicating with the text extraction system 102 and the data storage 104 via the communication network 108. The external devices 106 and the text extraction system 102 are generally disparately located.

[030] The functionalities of the external devices 106 may be implemented in portable devices, such as a high-speed computing device, and/or non-portable devices, such as an application server. Examples of the external devices 106 may include, but are not limited to, a computing device, a smart phone, a camera, a mobile device, a laptop, a personal digital assistant (PDA), a microphone, a printer and a tablet.

[031] The communication network 108 may include a communication medium through which the text extraction system 102, the data storage 104, and the external devices 106 may communicate with each other. Examples of the communication network 108 may include, but are not limited to, the Internet, a cloud network, a Wireless Fidelity (Wi-Fi) network, a Personal Area Network (PAN), a Local Area Network (LAN), or a Metropolitan Area Network (MAN). Various devices in the environment 100 may be configured to connect to the communication

network 108, in accordance with various wired and wireless communication protocols. Examples of such wired and wireless communication protocols may include, but are not limited to, a Transmission Control Protocol and Internet Protocol (TCP/IP), User Datagram Protocol (UDP), Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), Zig Bee, EDGE, IEEE 802.11, light fidelity(Li-Fi), 802.16, IEEE 802.11s, IEEE 802.11g, multi-hop communication, wireless access point (AP), device to device communication, cellular communication protocols, and Bluetooth (BT) communication protocols.

[032] The display unit 114 may be configured to communicate with different operational components of the text extraction system 102. The display unit 114 may be configured to display data for the text extraction system 102. A user may interact in the environment 100 via the UI 116 accessible via the display unit 114.

[033] During operation, the text extraction system 102 may be configured to receive an input from the external devices 106. The text extraction system 102 may be further configured to decipher a character pattern associated with the received input. For deciphering the character pattern, the text extraction system 102 may be configured to identify a character type from a plurality of character types associated with each of the plurality of characters in the received input. In accordance with an embodiment, the text extraction system 102 may be configured to assign, to each of the plurality of characters, a character code from a plurality of character codes based on the identified character type from the plurality of character types. In accordance with an embodiment, the text extraction system 102 may be further configured to create one or more clusters for the received input in response to assigning the character code to each of the plurality of characters. The one or more clusters may be stored in the memory 112 or the data storage 104, via the communication network 108. Each of the one or more clusters may include at least one contiguous occurrence of the same character code. In accordance with an embodiment, the text extraction system 102 may be configured to replace, for each of the one or more clusters, the at least one contiguous occurrence of the same character code with a single occurrence of the same character code to generate the character pattern. In accordance with an embodiment, the text extraction system 102 may be configured to extract, from the document, at least one text-entity matching the received input, based on the character pattern deciphered for the received input. The extracted at least one text entity may be displayed on the display unit 114 or the external devices 106 from the text extraction system 102, via the communication network 108.

[034] Additionally, in accordance with an embodiment, for easy extraction of text entities, the text extraction system 102 may be configured to determine a location identifier of the character pattern associated with each text-entity of the at least one text-entity within the document. In accordance with an embodiment, the character pattern may be localized based on the determination of the character pattern associated with each text-entity of the at least one text-entity within the document. Localization of the character pattern may correspond to determining the location of the character pattern in the document. In accordance with an embodiment, the location identifier may be an attribute of the document corresponding to the character pattern and may be included as metadata of the document for quick extraction of text entities.

[035] In accordance with an embodiment, the text extraction system 102 may be configured to create a mapping index, based on an association of the location identifier with the character pattern associated with each text-entity of the at least one text-entity within the document. In accordance with an embodiment, the text extraction system 102 may be configured to extract, from the document, at least one text-entity matching the received input, based on the mapping index.

[036] While example embodiments described herein generally relate to extracting text from a document based on a received input, example embodiments may also be implemented for extracting, without limitation, an icon, an image, an emoji, a logo, a barcode or a shape from the document, based on received input.

[037] All the components in the environment 100 may be coupled directly or indirectly to the communication network 108. The components described in the environment 100 may be further broken down into more than one component and/or combined together in any suitable arrangement. Further, one or more components may be rearranged, changed, added, and/or removed.

[038] FIG. 2 is a functional block diagram of a text extraction system for extracting text from a document based on a received input, in accordance with some other embodiments of the present disclosure. FIG. 2 is explained in conjunction with elements from FIG. 1.

[039] With reference to FIG. 2, there is shown a functional block diagram of a text extraction system 200 (corresponding to the text extraction system 102). The text extraction system 102 8ay include a model/rule engine 202, a confidence layer 204, a model output storage 206, an application feedback module 208, an inference layer 210, and a corpus/knowledge repository 212. The model/rule engine 202 may be communicatively coupled to the confidence layer 204,

the model output storage 206, the application feedback module 208, the inference layer 210, and the corpus/knowledge repository 212.

[040] Elements and features of the text extraction system 102 may be operatively associated with one another, coupled to one another, or otherwise configured to cooperate with one another as needed to support the desired functionality, as described herein. For ease of illustration and clarity, the various physical, electrical, and logical couplings and interconnections for the elements and the features are not depicted in FIG. 2. Moreover, it should be appreciated that embodiments of the text extraction system 102 will include other elements, modules, and features that cooperate to support the desired functionality. For simplicity, FIG. 2 only depicts certain elements that relate to the techniques described in more detail below.

[041] In some embodiments, the model/rule engine 202 may receive a set of feed files 214. By way of an example, a format for each of the set of feed files 214 may include, without limitation, a ".pdf" format, a ".txt" format, a ".doc" format, or a ".xlsx" format. It may be noted that the model/rule engine 202 may be developed based on the feed files 214.

[042] In some embodiments, the confidence layer 204 may be an extension to the model/rule engine 202. It may be noted that the confidence layer 204 may remove noise text-entities from the extracted text-entities matching the received input. Further, the confidence layer 204 may facilitate extraction of relevant text data.

[043] The model/rule engine 202 may generate a model output based on the feed files 214 which may be stored in the model output storage 206. Further, the model output storage 206 may be accessed by the application feedback module 208. It may be noted that the application feedback module 208 may receive an application feedback from a user as a response to the model output. In some embodiments, the application feedback module 208 may receive the application feedback from the user via the UI 116. Further, the application feedback may be stored in an application feedback output storage 216. Further, the application feedback may be received by the inference layer 210.

[044] In some embodiments, the inference layer 210 may identify a character pattern in the application feedback 208. Further, based on the pattern, the inference layer 210 may define a set of rules for the model/rule engine 202. The corpus/knowledge repository 212 may receive the character pattern or the set of rules, or both, from the inference layer 210. It may be noted that information stored in the corpus/knowledge repository 212 may be used by the model/rule engine 202 for extracting text-entities.

[045] In practice, the model/rule engine 202, the confidence layer 204, the application feedback module 208, and the inference layer 210 may be implemented with (or cooperate with) each other to perform at least some of the functions and operations described in more detail herein. In this regard, the model/rule engine 202, the confidence layer 204, the application feedback module 208, and the inference layer 210 may be realized as suitably written processing logic, application program code, or the like.

[046] FIG. 3 is a flowchart that illustrates an exemplary method 300 for identifying a character pattern in a document and generating a set of rules, in accordance with an embodiment of the present disclosure. FIG. 3 is explained in conjunction with elements from FIG. 1 and FIG.2. This method 300 may be executed by any computing system, for example, by the text extraction system 102 of FIG. 1.

[047] At step 302, an input may be received in form of an application feedback or a new suggestion may be received. In accordance with an embodiment, the inference layer 210 of the text extraction system 102 may be configured to receive the application feedback or the new suggestion from a user. At step 304, a set of character patterns may be uniquely identified for the feedback or a set of rules. In accordance with an embodiment, the inference layer 210 of the text extraction system 102 may be configured to uniquely identify the set of character patterns for the feedback or the set of rules. As such, the steps 302-304 may be performed by the inference layer 210.

[048] For example, in order to identify the set of character patterns for the feedback or the set of rules, a character type from a plurality of character types associated with each of the plurality of characters in the received input may be identified. A character code (from a plurality of character codes) may be assigned to each of the plurality of characters, based on the identified character type from the plurality of character types. One or more clusters may be created for the received input in response to assigning the character code to each of the plurality of characters, such that each of the one or more clusters includes at least one contiguous occurrence of the same character code. The at least one contiguous occurrence of the same character code may be replaced, for each of the one or more clusters, with a single occurrence of the same character code to generate the character pattern. Further, the set of character patterns may be encoded for unification. In accordance with an embodiment, the unification may be indicative of which set of a character pattern is same and which character pattern is different.

[049] At step 306, the set of character patterns may be transmitted to the corpus/knowledge repository 212. Further, at step 306, the set of character patterns may be checked for unique patterns. In accordance with an embodiment, once the set of patterns are transmitted to the corpus/knowledge repository 212, the set of character patterns may be checked for unique patterns, and redundant patterns may be deleted.

[050] Additionally, the text extraction system 102 may be configured to generate a set of rules using the inference layer 210. In an exemplary scenario, the text extraction system 102 may fail to extract/identify an oil well having a name "abc-123" from a document. Further, in this scenario, this oil well name may be identified manually by a user, via a user interface. Thereafter, the user may input this oil well name to the text extraction system 102 so as to train the text extraction system 102 to make it capable to extract in the future oil well names having a pattern similar to the pattern of this oil well name. As such, this oil well name may be received as input or feedback or a new suggestion

[051] Upon receiving the input, a character pattern associated with the input-token may be deciphered. Accordingly, a character type of each character in the input-token "abc-123" may be identified as "alphabet", "alphabet", "alphabet", "punctuation", "numerical digit", "numerical digit", "numerical digit", and character code may be assigned as "aaapddd". Further, one or more clusters may be created using similar character codes positioned adjacent to each other (i.e., contiguous occurrence of the same character code). Therefore, a first cluster of three "a" character codes, followed by a second cluster of single "p" character codes, and followed by a third cluster of three "d" character codes may be identified. Further, a cluster code may be assigned to the first, second, and third clusters to obtain the character pattern associated with the input-token. Therefore, the character pattern deciphered for the received input "abc-123" is "apd". The deciphering of the character pattern from the received input may also correspond to encoding.

[052] Further, the character pattern associated with the input-token may be mapped with the character patterns associated with the text-entities of the document to identify text-entities matching the input-token.

[053] Additionally, in accordance with an embodiment, for easy extraction of text entities, the text extraction system 102 may be configured to determine a location identifier of the character pattern associated with each text-entity of the at least one text-entity within the document. In accordance with an embodiment, the character pattern may be localized based on the determination of the character pattern associated with each text-entity of the at least one text-

entity within the document. Localization of the character pattern may correspond to determining the location of the character pattern in the document. In accordance with an embodiment, the location identifier may be an attribute of the document corresponding to the character pattern and may be included as metadata of the document for quick extraction of text entities. In accordance with an embodiment, the text extraction system 102 may be configured to create a mapping index, based on an association of the location identifier with the character pattern associated with each text-entity of the at least one text-entity within the document. In accordance with an embodiment, the text extraction system 102 may be configured to extract, from the document, at least one text-entity matching the received input, based on the mapping index.

[054] In continuation of the example above, each of the text-entities in the document of the pattern "apd" may be extracted. In order to extract, from a document, text-entities matching the oil well name, in some embodiments, the character pattern associated with the oil well name, i.e., "apd" may be decoded. Accordingly, character patterns associated with text-entities of the document may be deciphered. Once the character patterns associated with the text-entities are deciphered, the character pattern "apd" may be mapped with the character patterns associated with the text-entities of the document to identify text-entities matching the oil well name. In other words, all the text-entities having a character pattern matching with the pattern "apd" may be extracted.

[055] Referring now to FIG. 4, a block diagram of a model/rule engine 202 of a text extraction system for extracting a character pattern in a document based on a set of rules is illustrated, in accordance with an embodiment of the present disclosure. FIG. 4 is explained in conjunction with elements from FIG. 1 to FIG. 3

[056] The model/rule engine 202 may receive an encoded regex pattern from an encoded regex pattern repository 402. The encoded regex pattern repository 402 may include character patterns associated with each text-entity within a document. In accordance with an embodiment, a rules generation module 406 of the model/rule engine 202 may be configured to receive the encoded regex pattern from the encoded regex pattern repository 402. Further, in accordance with an embodiment, the rules generation module 406 may be configured to generate a set of rules by decoding the encoded patterns, based on the encoded regex pattern.

[057] A relevant data extraction model 408 of the model/rule engine 202 may receive the set of rules from the rules generation module 406. The rules generation module 406 may decode the encoded patterns (i.e., encoding performed via the method 300) which may help in

extracting text entities. Further, the relevant data extraction model 408 may receive feed files 214. It may be noted that the relevant data extraction model 408 may extract a value from the feed files 214 based on the set of rules. The extracted value may later be stored in the extracted value storage 404.

[058] FIG. 5 is a flowchart that illustrates an exemplary method 500 of extracting text from a document based on a received input, in accordance with an embodiment of the present disclosure. FIG. 5 is explained in conjunction with elements from FIG. 1 to FIG. 4. The operations of the exemplary method 500 may be executed by any computing system, for example, by the text extraction system 102 of FIG. 1. The operations of the method 500 may start at step 502 and proceed to step 504.

[059] At step 502, a character pattern associated with the received input may be deciphered. In accordance with an embodiment, the text extraction system 102 may be configured to decipher a character pattern associated with the received input. In accordance with an embodiment, the received input (also referred as input-token) may include a plurality of characters. For example, the character type associated with each of the plurality of characters in the received input may include at least one of: an alphabet, a punctuation, or a digit. The step 502 of deciphering the character pattern associated with the received input is further explained in detail in conjunction with FIG. 6.

[060] Referring now to FIG. 6, a flowchart of an exemplary method 600 of deciphering a character pattern associated with the received input is illustrated, in accordance with an embodiment of the present disclosure. FIG. 6 is explained in conjunction with elements from FIG. 1 to FIG. 5.

[061] At step 602, a character type from a plurality of character types may be identified associated with each of the plurality of characters in the received input. In accordance with an embodiment, the text extraction system 102 may be configured to identify a character type from a plurality of character types associated with each of the plurality of characters in the received input.

[062] At step 604, a character code from a plurality of character codes may be assigned to each of the plurality of characters. In accordance with an embodiment, the text extraction system 102 may be configured to assign, to each of the plurality of characters, a character code from a plurality of character codes based on the identified character type from the plurality of character types. For example, the character code corresponding to an alphabet may be 'a'. Similarly, the character code corresponding to a punctuation may be 'p', and to a digit may be 'd'.

[063] At step 606, one or more clusters may be created for the received input. In accordance with an embodiment, the text extraction system 102 may be configured to create one or more clusters for the received input in response to assigning the character code to each of the plurality of characters. In accordance with an embodiment, each of the one or more clusters may include the at least one contiguous occurrence of the same character code.

[064] At step 608, for each of the one or more clusters, the at least one contiguous occurrence of the same character code may be replaced with a single occurrence of the same character code to generate the character pattern. In accordance with an embodiment, the text extraction system 102 may be configured to replace, for each of the one or more clusters, the at least one contiguous occurrence of the same character code with the single occurrence of the same character code to generate the character pattern.

[065] Returning to FIG. 5, at step 504, at least one text-entity from the document may be extracted that matches the received input. In accordance with an embodiment, the text extraction system 102 may be configured to extract, from the document, the at least one text-entity matching the received input, based on the character pattern deciphered for the received input. It may be noted that once the character pattern associated with the received input (or input-token) may be deciphered, the character pattern may be used to extract, from the document, one or more text-entities which match the pattern associated with the input-token.

[066] Referring now to FIG. 7, a flowchart of an exemplary method 700 of extracting, from a document, at least one text-entity matching the received input is illustrated, in accordance with an embodiment of the present disclosure. FIG. 7 is explained in conjunction with elements from FIG. 1 to FIG. 6. The operations of the exemplary method 700 may be executed by any computing system, for example, by the text extraction system 102.

[067] At step 702, a character pattern associated with each text-entity of the at least one text-entity may be deciphered from the document. In accordance with an embodiment, the text extraction system 102 may be configured to decipher the character pattern associated with each text-entity of the at least one text-entity from the document. For example, the document may be parsed to obtain various text-entities like words, in the document. In accordance with an embodiment, the character type associated with each of the plurality of characters in each text-entity of the document may include at least one of: an alphabet, a punctuation, or a digit. The step 702 of deciphering a character pattern associated with each text-entity of the at least one text-entity from the document is further explained in conjunction with FIG. 8.

[068] Referring now to FIG. 8, a flowchart of an exemplary method 800 of deciphering a character pattern associated with each text-entity of the at least one text-entity from the document is illustrated, in accordance with an embodiment of the present disclosure. FIG. 8 is explained in conjunction with elements from FIG. 1 to FIG. 7. The operations of the exemplary method 800 may be executed by any computing system, for example, by the text extraction system 102.

[069] At step 802, a character type from a plurality of character types associated with each of a plurality of characters may be identified in each text-entity of the document. In accordance with an embodiment, the text extraction system 102 may be configured to identify the character type from the plurality of character types associated with each of the plurality of characters in each text-entity of the document. For example, the character type associated with text characters of each text-entity may include an alphabet, or a punctuation, or a digit.

[070] At step 804, a character code from a plurality of character codes may be assigned to each of the plurality of characters. In accordance with an embodiment, the text extraction system 102 may be configured to assign, to each of the plurality of characters, the character code from the plurality of character codes based on the identified character type from the plurality of character types. For example, the character code corresponding to an alphabet may be 'a', the character code corresponding to a punctuation may be 'p', and to a digit may be 'd'.

[071] At step 806, one or more clusters may be created for the document. In accordance with an embodiment, the text extraction system 102 may be configured to create one or more clusters for the document in response to assigning the character code to each of the plurality of characters. In accordance with an embodiment, each of the one or more clusters may include at least one contiguous occurrence of the same character code. In other words, one or more clusters may be created using similar character codes positioned adjacent to each other.

[072] At step 808, at least one contiguous occurrence of the same character code may be replaced with a single occurrence of the same character code to generate the character pattern. In accordance with an embodiment, the text extraction system 102 may be configured to replace, for each of the one or more clusters, the at least one contiguous occurrence of the same character code with the single occurrence of the same character code to generate the character pattern.

[073] Returning to FIG. 7, at step 704, character pattern deciphered from the received input may be mapped with the character pattern associated with each of the at least one text-entity from the document to extract at least one text-entity from the document matching the received

input. In accordance with an embodiment, the text extraction system 102 may be configured to map the character pattern deciphered from the received input with the character pattern associated with each of the at least one text-entity from the document to extract at least one text-entity from the document matching the received input.

[074] Referring not to FIG. 9, a flowchart of an exemplary method 900 of removing at least one noise text-entity from the extracted at least one text-entity matching the received input is illustrated, in accordance with an embodiment of the present disclosure. FIG. 9 is explained in conjunction with elements from FIG. 1 to FIG. 8. The operations of the exemplary method 900 may be executed by any computing system, for example, by the text extraction system 102.

[075] At step 902, a reference text-entity having a semantic or a syntactic relationship with the received input may be identified from the document. In accordance with an embodiment, the text extraction system 102 may be configured to identify a reference text-entity having a semantic or a syntactic relationship with the received input from the document. At step 904, a distance of each of the extracted at least one text-entity may be determined from the reference text-entity. In accordance with an embodiment, the text extraction system 102 may be configured to determine the distance of each of the extracted at least one text-entity from the reference text-entity.

[076] At step 906, a weight may be assigned, to each of the at least one extracted text-entity, based on the distance of each of the extracted at least one text-entity from the reference text-entity. In accordance with an embodiment, the text extraction system 102 may be configured to assign, to each of the at least one extracted text-entity, the weight, based on the distance of each of the extracted at least one text-entity from the reference text-entity. By way of an example, the weight may be inversely proportional to the distance, i.e., text-entity nearer to the reference text-entity may carry higher weight. In accordance with an embodiment, the text extraction system 102 may be configured to calculate the weight for each of the at least one text-entity. In accordance with an embodiment, the weight may be calculated based on a product of the distance of the one of the at least one text entity from the reference text-entity and a similarity of the one of the at least one text entity with the reference text-entity. For example, a weight value (also called confidence value) may be calculated, such that the weight value is a function of distance (a distance value) and similarity (a match value), as given below:

$$\text{Weight Value} = (\text{distance value}) * (\text{match value})$$

[077] It may be noted that the highest weight value may be 1 and the lowest weight value may be 0. For example, if a keyword is at 10th position and the match value (or a probability of

match) is 0.85, then the weight value (also referred to as confidence value) may be calculated as below:

Weight Value = 1*0.85 = 0.85

[078] At step 908, one or more text-entities may be selected from the extracted at least one text-entity, based on the weight. In accordance with an embodiment, the text extraction system 102 may be configured to selecting one or more text-entities from the extracted at least one text-entity, based on the weight. In this way, text-entities lying farther away from the reference text-entity and, therefore, having lower relevance with the received input (or input-token) may be removed.

[079] For example, the confidence layer 204 of the text extraction system 102 may be configured to remove one or more noise text-entities from the extracted text-entities matching the received input (or input-token), and narrow them down to legitimate text-entities. The confidence layer 204 of the text extraction system 102 may be configured to calculate confidence associated with each text-entity based on parameters, like, distance of the reference text-entity (keyword) from an actual value (that is, how far the keywords are from actual value).

[080] By way of an example, the extent of keywords is searched 10 words left and right of "VALUE". The reference text-entity "VALUE" may occur be in a document as shown below:

"X1 X2 X3 X4 X5 X6 X7 X8 X9 X10 VALUE Y1 Y2 Y3 Y4 Y5 Y6 Y7 Y8 Y9 Y10"

[081] Distance of each of the extracted text-entities may be determined from the reference text-entity "VALUE". It may be understood that the distance of text-entity "X10" and text-entity "Y1" from the reference text-entity "VALUE" will be same = 10/10 = 1 (It is 10^{th} position out of 10, which is the nearest to "VALUE"). The value for X9 and Y2 will be same = 9/10 =0.9 (since, it is 9^{th} position out of 10). Similarly, the distance of text-entity "X6" from the reference text-entity "VALUE" is 5.

[082] Based on the distance, a weight may be assigned to each of the extracted text-entities. It may be understood that weight may be inversely proportional to the distance, i.e., text-entity nearer to the reference text-entity may carry higher weight.

[083] In accordance with an embodiment, repetition may be used as one of a confidence types, without limitation. The repetition may correspond to a frequency-based scoring algorithm that checks the number of occurrences of a value in the document found with respect to the character

patterns provided through the inference layer 210. Repetition value may be calculated by using formula given below:

[084] Repetition = Occurrences in document / Total occurrences of values extracted from document

[085] By way of an example, five patterns may be stored in memory 112 of the text extraction system 102, and the text extraction system 102 may extract two unique values using the five patterns. Further, for example, value "1" is found in the document 15 times and Value "2" is found in the document 24 times. Therefore,

Confidence score for Value "1" = 15 / (15+24) = 0.385.

Confidence Score for Value "2" = 24 / (15+24) = 0.62

[086] In accordance with another embodiment, page number may be used as another confidence type, without limitation. In accordance with an embodiment, a segment may be scored, based on page number as given below:

Page weightage = (1-page number/total pages + 1/total pages)

[087] By way of an example, when an attribute is present in the second page and the document has 5 pages, then,

Page weightage for that attribute = (1- 2/5 + 1/5) = 0.8

[088] One or more techniques of extracting, from a document, text entities text-entities matching a received input are disclosed. The disclosed techniques provide for time and labor efficient training of a rule engine for improving the accuracy of extracting text-entities from the document. The disclosed text extraction system further facilitates identification of various different type of relevant text-entities in a document, while minimizing chances of missing any relevant text-entities. Further, by using weight/confidence scoring algorithms, the techniques provide for eliminating or minimizing chances of extracting noise results.

[089] Furthermore, one or more computer-readable storage media may be utilized in implementing embodiments consistent with the present disclosure. A computer-readable storage medium refers to any type of physical memory on which information or data readable by a processor may be stored. Thus, a computer-readable storage medium may store instructions for execution by one or more processors, including instructions for causing the processor(s) to perform steps or stages consistent with the embodiments described herein. The term "computer-readable medium" should be understood to include tangible items and exclude

carrier waves and transient signals, i.e., be non-transitory. Examples include random access memory (RAM), read-only memory (ROM), volatile memory, nonvolatile memory, hard drives, CD ROMs, DVDs, flash drives, disks, and any other known physical storage media.

[090] It will be appreciated that, for clarity purposes, the above description has described embodiments with reference to different functional units and processors. However, it will be apparent that any suitable distribution of functionality between different functional units, processors or domains may be used without detracting from the disclosure. For example, functionality illustrated to be performed by separate processors or controllers may be performed by the same processor or controller. Hence, references to specific functional units are only to be seen as references to suitable means for providing the described functionality, rather than indicative of a strict logical or physical structure or organization.

[091] Although the present disclosure has been described in connection with some embodiments, it is not intended to be limited to the specific form set forth herein. Rather, the scope of the present disclosure is limited only by the claims. Additionally, although a feature may appear to be described in connection with particular embodiments, one skilled in the art would recognize that various features of the described embodiments may be combined in accordance with the disclosure.

[092] Furthermore, although individually listed, a plurality of means, elements or process steps may be implemented by, for example, a single unit or processor. Additionally, although individual features may be included in different claims, these may possibly be advantageously combined, and the inclusion in different claims does not imply that a combination of features is not feasible and/or advantageous. Also, the inclusion of a feature in one category of claims does not imply a limitation to this category, but rather the feature may be equally applicable to other claim categories, as appropriate.

[093] It is intended that the disclosure and examples be considered as exemplary only, with a true scope and spirit of disclosed embodiments being indicated by the following claims.

We claim:

1. A method (500) for extracting text from a document based on a received input, the method comprising:

deciphering (502) a character pattern associated with the received input, wherein the received input comprises a plurality of characters wherein deciphering the character pattern comprises:

identifying (602) a character type from a plurality of character types associated with each of the plurality of characters in the received input;

assigning (604), to each of the plurality of characters, a character code from a plurality of character codes based on the identified character type from the plurality of character types;

creating (606) one or more clusters for the received input in response to assigning the character code to each of the plurality of characters, wherein each of the one or more clusters comprises at least one contiguous occurrence of the same character code; and

replacing (608), for each of the one or more clusters, the at least one contiguous occurrence of the same character code with a single occurrence of the same character code to generate the character pattern; and

extracting (504), from the document, at least one text-entity matching the received input, based on the character pattern deciphered for the received input.

2. The method (500) as claimed in claim 1, wherein extracting (504) the at least one text-entity from the document further comprises:

deciphering (702) a character pattern associated with each text-entity of the at least one text-entity from the document, wherein deciphering the character pattern comprises:

identifying (802) a character type from a plurality of character types associated with each of a plurality of characters in each text-entity of the document;

assigning (804), to each of the plurality of characters, a character code from a plurality of character codes based on the identified character type from the plurality of character types;

creating (806) one or more clusters for the document in response to assigning the character code to each of the plurality of characters, wherein each of the one or more clusters comprises at least one contiguous occurrence of the same character code; and

replacing (808), for each of the one or more clusters, at least one contiguous occurrence of the same character code with a single occurrence of the same character code to generate the character pattern; and

mapping (704) the character pattern deciphered from the received input with the character pattern associated with each of the at least one text-entity from the document to extract at least one text-entity from the document matching the received input.

3. The method (500) as claimed in claim 2, wherein the character type associated with each of the plurality of characters in the received input or the character type associated with each of the plurality of characters in each text-entity of the document comprises at least one of: an alphabet, a punctuation, or a digit.

4. The method (500) as claimed in claim 1, further comprising removing at least one noise text-entity from the extracted at least one text-entity matching the received input, wherein removing the at least one noise text-entity comprises:

identifying (902) from the document, a reference text-entity having a semantic or syntactic relationship with the received input;

determining (904) a distance of each of the extracted at least one text-entity from the reference text-entity;

assigning (906), to each of the at least one extracted text-entity, a weight, based on the distance of each of the extracted at least one text-entity from the reference text-entity; and

selecting (908) one or more text-entities from the extracted at least one text-entity, based on the weight.

5. The method (500) as claimed in claim 4, further comprising calculating the weight for each of the at least one text-entity, wherein the weight is calculated based on a product of the distance of the one of the at least one text entity from the reference text-entity and a similarity of the one of the at least one text entity with the reference text-entity.

6. The method (500) as claimed in claim 1, further comprising:

determining a location identifier of the character pattern associated with each text-entity of the at least one text-entity within the document;

creating a mapping index, based on an association of the location identifier with the character pattern associated with each text-entity of the at least one text-entity within the document; and

extracting, from the document, at least one text-entity matching the received input, based on the mapping index.

7. A system for extracting text from a document based on a received input, the system comprising:

a processor (110); and

a memory (112) communicatively coupled to the processor (110), wherein the memory (112) stores processor-executable instructions, which, on execution by the processor (110), cause the processor (110) to:

decipher a character pattern associated with the received input, wherein the received input comprises a plurality of characters wherein deciphering the character pattern comprises:

identifying a character type from a plurality of character types associated with each of the plurality of characters in the received input;

assigning, to each of the plurality of characters, a character code from a plurality of character codes based on the identified character type from the plurality of character types;

creating one or more clusters for the received input in response to assigning the character code to each of the plurality of characters, wherein each of the one or more clusters comprises at least one contiguous occurrence of the same character code; and

replacing, for each of the one or more clusters, the at least one contiguous occurrence of the same character code with a single occurrence of the same character code to generate the character pattern; and

extract, from the document, at least one text-entity matching the received input, based on the character pattern deciphered for the received input.

8. The system as claimed in claim 7, wherein extracting the at least one text-entity from the document further comprises:

deciphering a character pattern associated with each text-entity of the at least one text-entity from the document, wherein deciphering the character pattern comprises:

identifying a character type from a plurality of character types associated with each of a plurality of characters in each text-entity of the document;

assigning, to each of the plurality of characters, a character code from a plurality of character codes based on the identified character type from the plurality of character types;

creating one or more clusters for the document t in response to assigning the character code to each of the plurality of characters, wherein each of the one or more clusters comprises at least one contiguous occurrence of the same character code; and

replacing, for each of the one or more clusters, at least one contiguous occurrence of the same character code with a single occurrence of the same character code to generate the character pattern; and

mapping the character pattern deciphered from the received input with the character pattern associated with each of the at least one text-entity from the document to extract at least one text-entity from the document matching the received input.

9. The system as claimed in claim 8, wherein the character type associated with each of the plurality of characters in the received input or the character type associated with each of the plurality of characters in each text-entity of the document comprises at least one of: an alphabet, a punctuation, or a digit.

10. The system as claimed in claim 7, wherein the processor-executable instructions, on execution by the processor, further cause the processor to remove at least one noise text-entity from the extracted at least one text-entity matching the received input, wherein removing the at least one noise text-entity comprises:

identifying from the document, a reference text-entity having a semantic or syntactic relationship with the received input;

determining a distance of each of the extracted at least one text-entity from the reference text-entity;

assigning, to each of the at least one extracted text-entity, a weight, based on the distance of each of the extracted at least one text-entity from the reference text-entity; and

selecting one or more text-entities from the extracted at least one text-entity, based on the weight.

11. The system as claimed in claim 10, wherein the processor-executable instructions, on execution by the processor, further cause the processor to calculate the weight for each of the at least one text-entity, wherein the weight is calculated based on a product of the distance of the one of the at least one text entity from the reference text-entity and a similarity of the one of the at least one text entity with the reference text-entity.

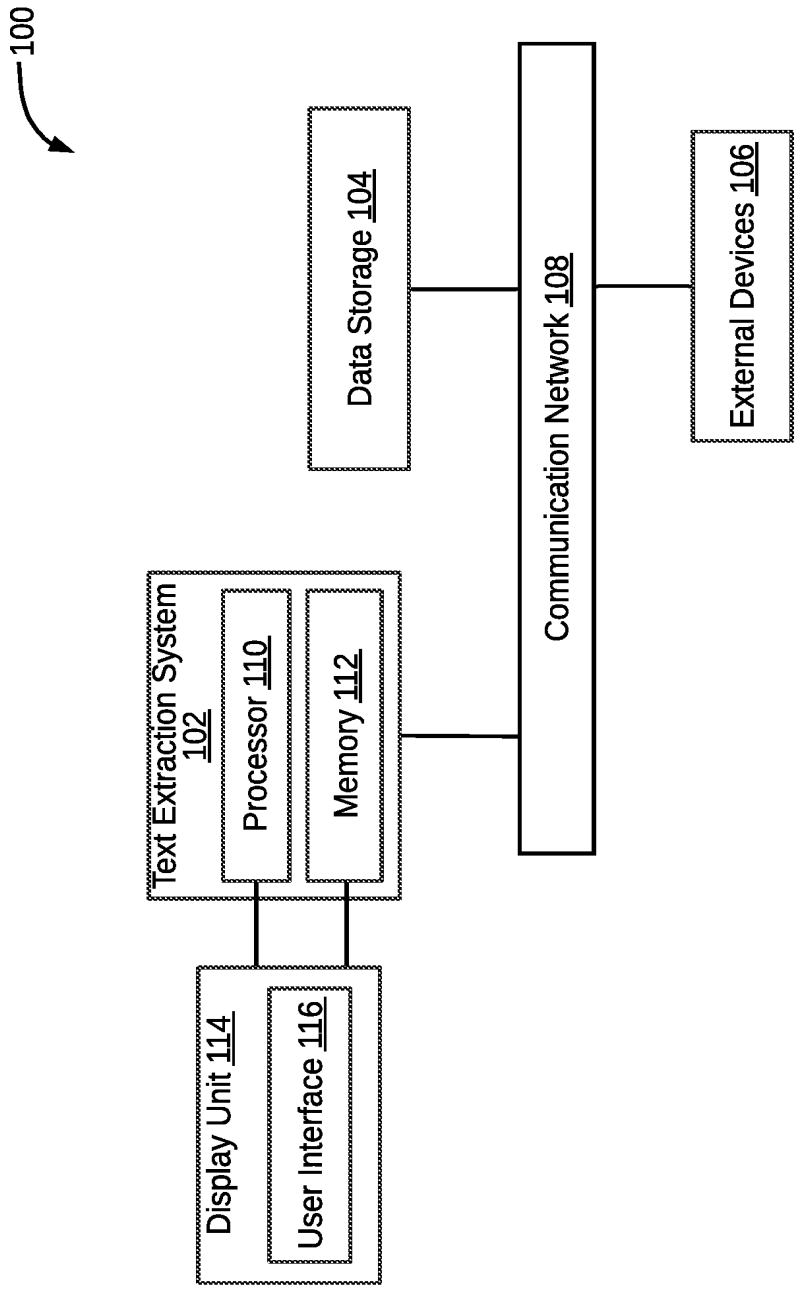

100
Data Storage 104
External Devices 106
Communication Network 108
Text Extraction System 102
Processor 110
Memory 112
Display Unit 114
User Interface 116

FIG. 1

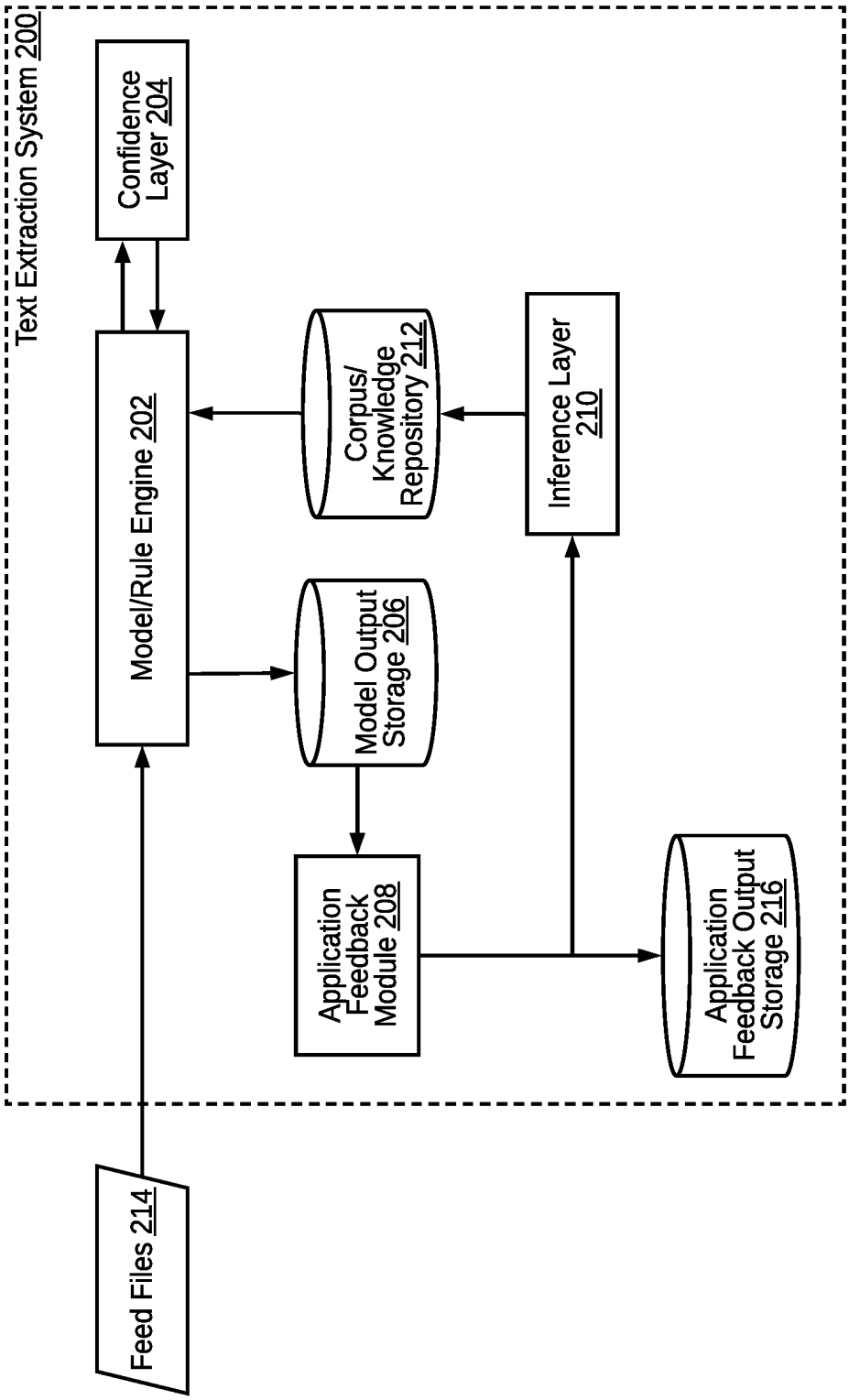

Text Extraction System 200
Confidence Layer 204
Model/Rule Engine 202
Corpus/ Knowledge Repository 212
Inference Layer 210
Model Output Storage 206
Application Feedback Module 208
Application Feedback Output Storage 216
Feed Files 214

FIG. 2

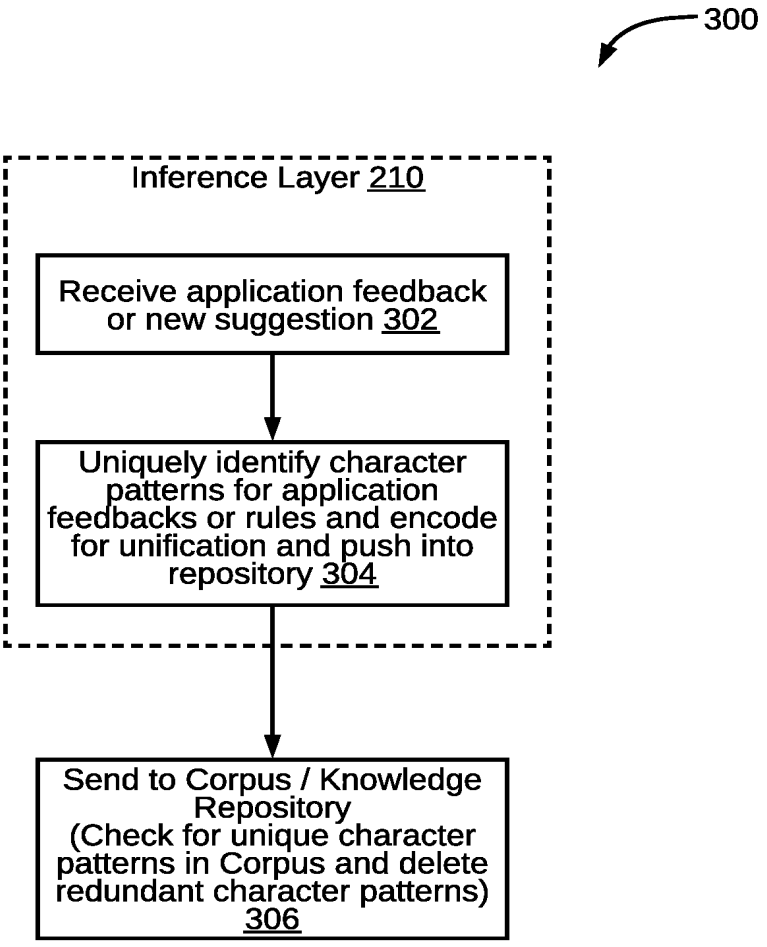

300
Inference Layer 210
Receive application feedback or new suggestion 302
Uniquely identify character patterns for application feedbacks or rules and encode for unification and push into repository 304
Send to Corpus / Knowledge Repository (Check for unique character patterns in Corpus and delete redundant character patterns) 306

FIG. 3

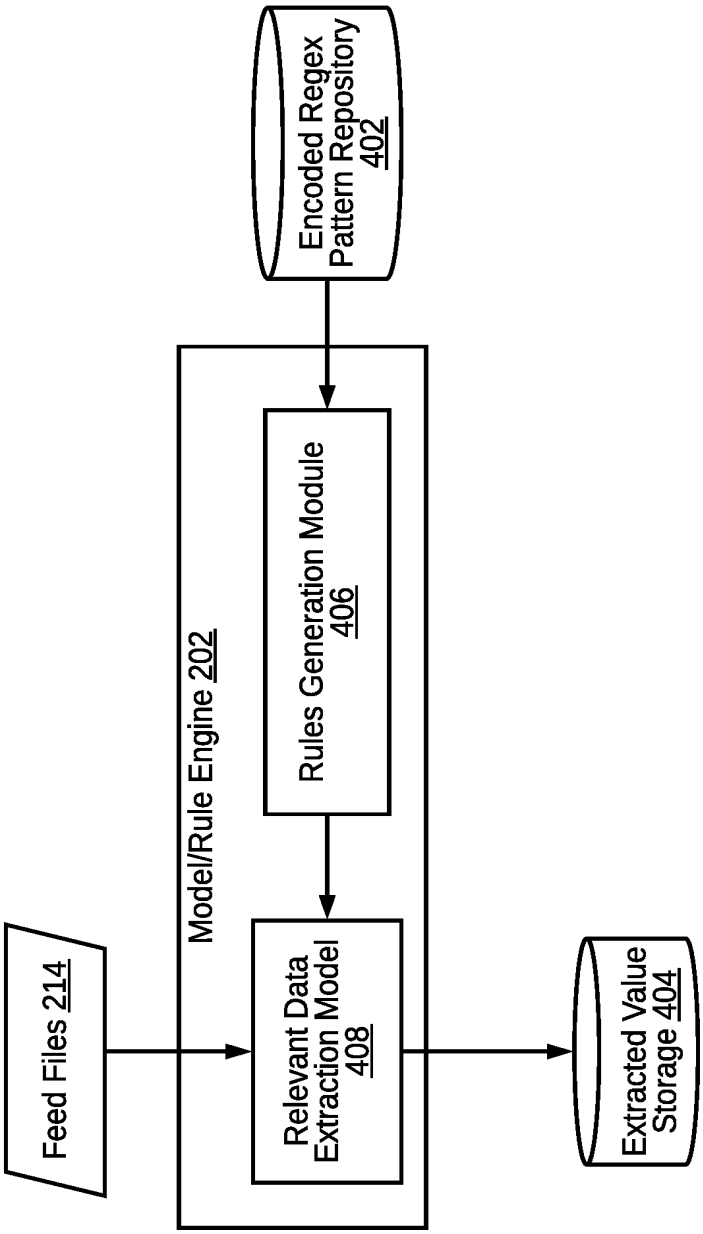

Encoded Regex Pattern Repository 402
Model/Rule Engine 202
Rules Generation Module 406
Feed Files 214
Relevant Data Extraction Model 408
Extracted Value Storage 404

FIG. 4

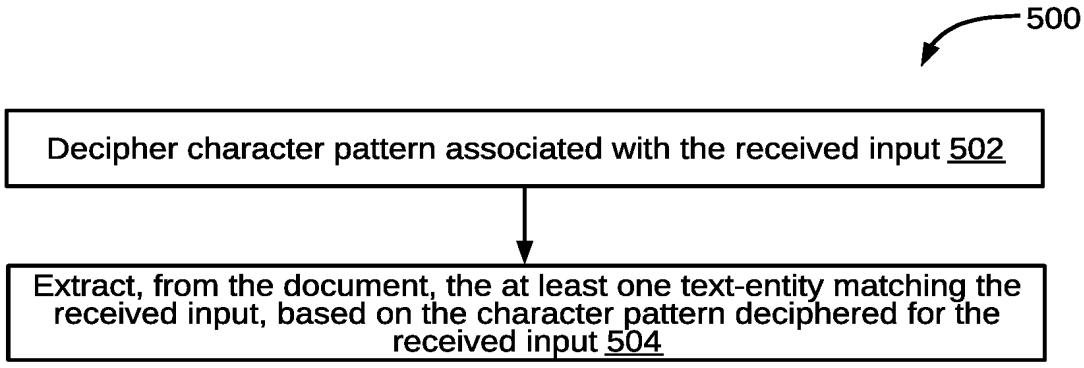

500
Decipher character pattern associated with the received input 502
Extract, from the document, the at least one text-entity matching the received input, based on the character pattern deciphered for the received input 504

FIG. 5

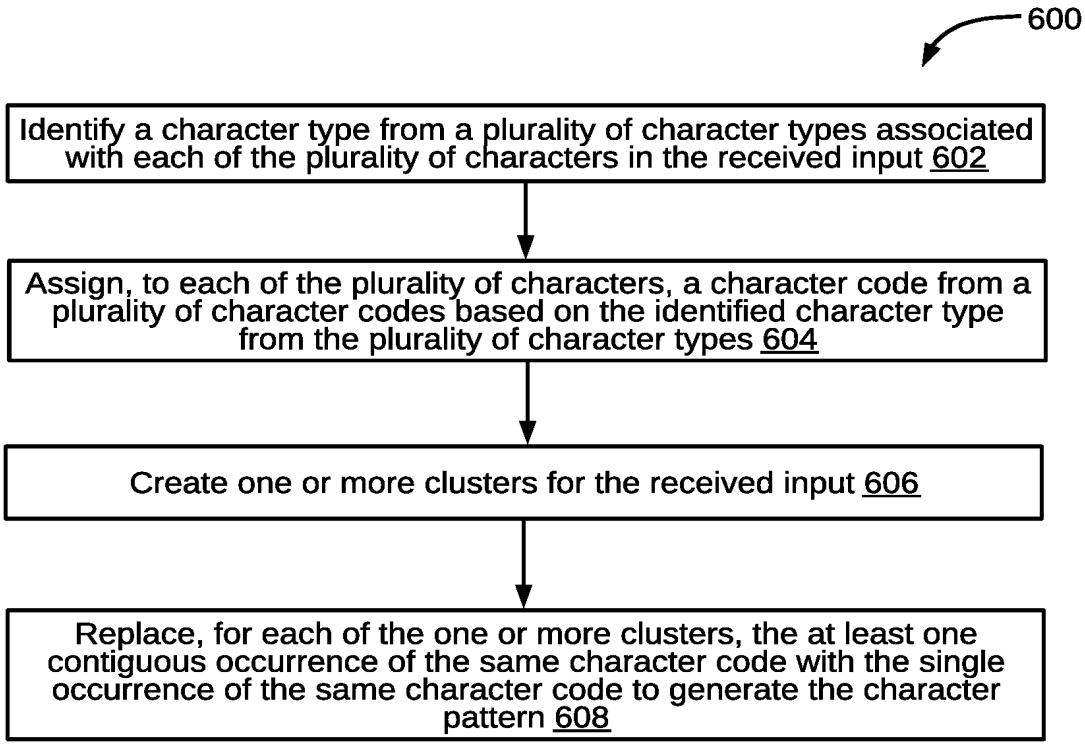

600
Identify a character type from a plurality of character types associated with each of the plurality of characters in the received input 602
Assign, to each of the plurality of characters, a character code from a plurality of character codes based on the identified character type from the plurality of character types 604
Create one or more clusters for the received input 606
Replace, for each of the one or more clusters, the at least one contiguous occurrence of the same character code with the single occurrence of the same character code to generate the character pattern 608

FIG. 6

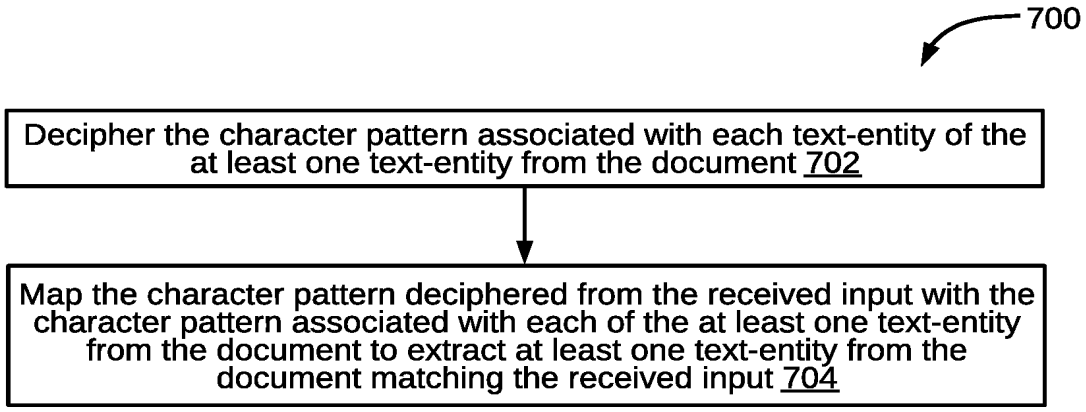

700
Decipher the character pattern associated with each text-entity of the at least one text-entity from the document 702
Map the character pattern deciphered from the received input with the character pattern associated with each of the at least one text-entity from the document to extract at least one text-entity from the document matching the received input 704

FIG. 7

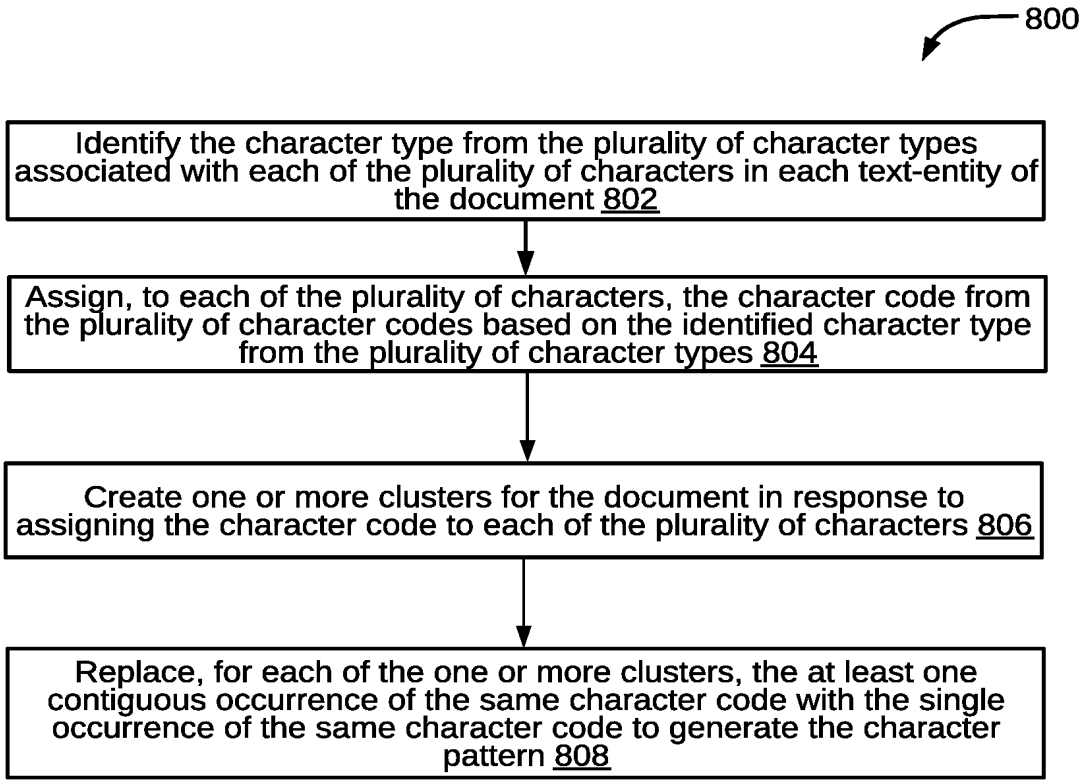

800
Identify the character type from the plurality of character types associated with each of the plurality of characters in each text-entity of the document 802
Assign, to each of the plurality of characters, the character code from the plurality of character codes based on the identified character type from the plurality of character types 804
Create one or more clusters for the document in response to assigning the character code to each of the plurality of characters 806
Replace, for each of the one or more clusters, the at least one contiguous occurrence of the same character code with the single occurrence of the same character code to generate the character pattern 808

FIG. 8

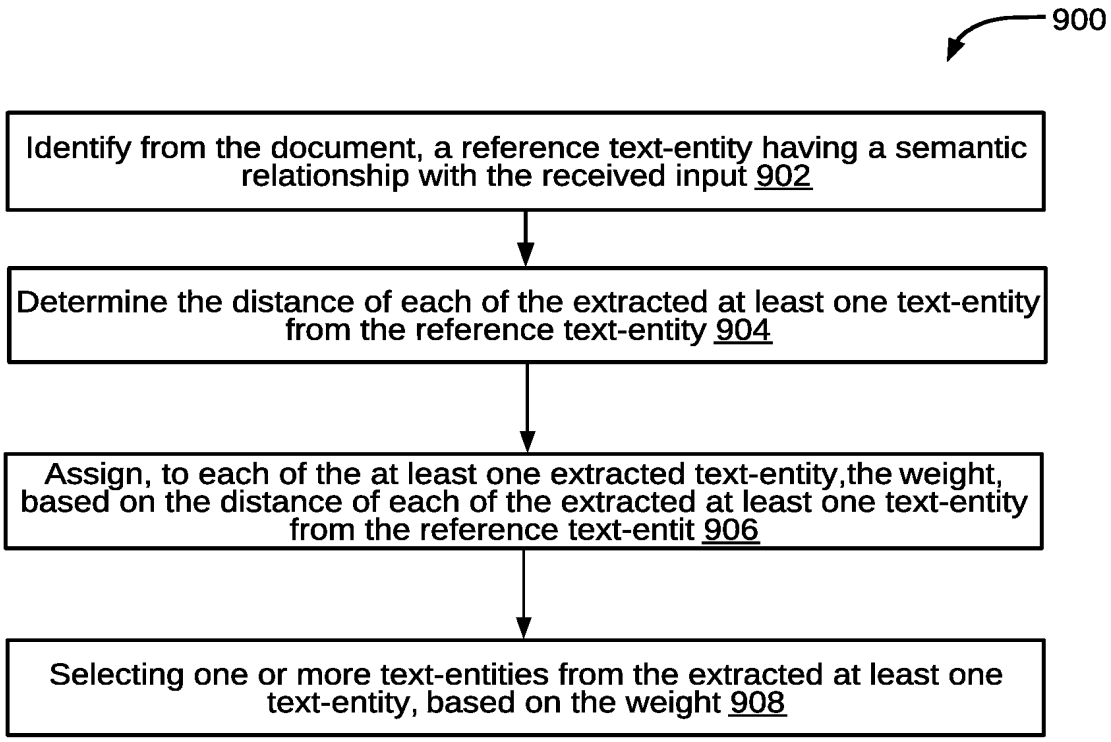

900
Identify from the document, a reference text-entity having a semantic relationship with the received input 902
Determine the distance of each of the extracted at least one text-entity from the reference text-entity 904
Assign, to each of the at least one extracted text-entity,the weight, based on the distance of each of the extracted at least one text-entity from the reference text-entit 906
Selecting one or more text-entities from the extracted at least one text-entity, based on the weight 908

FIG. 9

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IB2021/052233

A. CLASSIFICATION OF SUBJECT MATTER
G06F16/20,G06F16/30,G06F17/00,G06F40/00 Version=2021.01

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

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

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

Databases: TotalPatent One, Google Patent
Keywords: text search, decipher, character pattern

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO2019241422A1 (ORACLE INTERNATIONAL CORPORATION) 19 DECEMBER 2019 (19-12-2019) whole document	1-11

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"D" document cited by the applicant in the international application
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
24-06-2021	24-06-2021

Name and mailing address of the ISA/	Authorized officer
Indian Patent Office Plot No.32, Sector 14,Dwarka,New Delhi-110075 Facsimile No.	Anil Tagale Telephone No. +91-1125300200

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/IB2021/052233

Citation	Pub.Date	Family	Pub.Date
WO 2019/241422 A1	19-12-2019	EP 3807787 A1	21-04-2021
		US 20190384783 A1	19-12-2019
		CN 112262390 A	22-01-2021