



(51) International Patent Classification:

G06F 16/84 (2019.01) G06F 16/332 (2019.01)

(21) International Application Number:

PCT/US2021/034689

(22) International Filing Date:

28 May 2021 (28.05.2021)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

202011021589.5 25 September 2020 (25.09.2020) CN

(71) Applicant: **MICROSOFT TECHNOLOGY LICENSING, LLC** [US/US]; One Microsoft Way, Redmond, Washington 98052-6399 (US).

(72) Inventors: **SHOU, Linjun**; MICROSOFT TECHNOLOGY LICENSING, LLC, One Microsoft Way, Redmond, Washington 98052-6399 (US). **GONG, Ming**; MICROSOFT TECHNOLOGY LICENSING, LLC, One Microsoft Way, Redmond, Washington 98052-6399 (US). **WANG, Xueyun**; MICROSOFT TECHNOLOGY LICENSING, LLC, One Microsoft Way, Redmond, Washington 98052-6399 (US). **JIANG, Daxin**; MICROSOFT

TECHNOLOGY LICENSING, LLC, One Microsoft Way, Redmond, Washington 98052-6399 (US).

(74) Agent: **SWAIN, Cassandra T.** et al.; MICROSOFT TECHNOLOGY LICENSING, LLC, One Microsoft Way, Redmond, Washington 98052-6399 (US).

(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, 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,

(54) Title: REPRESENTATION LEARNING OF SEMI-STRUCTURED DATA

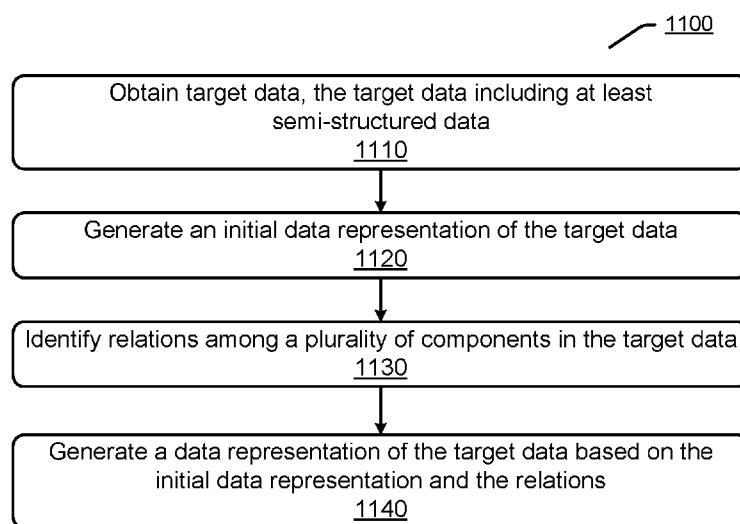


FIG 11

(57) Abstract: The present disclosure proposes method and apparatus for obtaining a representation of target data. Target data may be obtained, the target data including at least semi-structured data. An initial data representation of the target data may be generated. Relations among a plurality of components in the target data may be identified. A data representation of the target data may be generated based on the initial data representation and the relations.

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 applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*
- *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*

Published:

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

REPRESENTATION LEARNING OF SEMI-STRUCTURED DATA

BACKGROUND

[0001] Question Answering (QA) is an important function of search engines, which can provide users with desired information in an efficient and easy-to-understand manner. For example, after receiving a question posed by a user, a search engine may use, e.g., information extracted from the Web to provide the user with an answer to the question. The information extracted from the Web includes, e.g., text paragraphs, semi-structured data, etc. The semi-structured data may refer to data that does not fully meet representation and storage requirements of a relational database but contains tags for separating semantic elements and layering fields. The semi-structured data may be data between structured data and unstructured data, wherein the structured data refers to data that may be represented and stored using a relational database and expressed through a two-dimensional table structure, while the unstructured data refers to data whose data structure is irregular or incomplete, inconvenient to use a predefined data model to organize, and inconvenient to express through a two-dimensional table structure. The semi-structured data may include, e.g., a table, a list, etc. Compared with unstructured plain text, the semi-structured data has an intrinsic structure, which can carry semantic relations among various elements of the semi-structured data. The semantic relations hidden in the structure of the semi-structured data are useful for answering user questions. A large amount of semi-structured data existing in the Web may be used as a rich information resource for realizing the Question Answering function. For example, for a question "Which place is best for skydiving?", a list extracted from the Web containing places in order that are suitable for skydiving may be used to answer the question.

SUMMARY

[0002] This Summary is provided to introduce a selection of concepts that are further described below in the Detailed Description. It is not intended to identify key features or essential features of the claimed subject matter, nor is it intended to be used to limit the scope of the claimed subject matter.

[0003] Embodiments of the present disclosure propose method and apparatus for obtaining a representation of target data. Target data may be obtained, the target data including at least semi-structured data. An initial data representation of the target data may be generated. Relations among a plurality of components in the target data may be identified. A data representation of the target data may be generated based on the initial

data representation and the relations.

[0004] It should be noted that the above one or more aspects comprise the features hereinafter fully described and particularly pointed out in the claims. The following description and the drawings set forth in detail certain illustrative features of the one or more aspects. These features are only indicative of the various ways in which the principles of various aspects may be employed, and this disclosure is intended to include all such aspects and their equivalents.

BRIEF DESCRIPTION OF THE DRAWINGS

[0005] The disclosed aspects will hereinafter be described in connection with the appended drawings that are provided to illustrate and not to limit the disclosed aspects.

[0006] FIG.1 illustrates exemplary semi-structured data.

[0007] FIG.2 illustrates an exemplary process for constructing a graph corresponding to target data according to an embodiment of the present disclosure.

[0008] FIG.3 illustrates exemplary graphs according to an embodiment of the present disclosure.

[0009] FIG.4 illustrates another exemplary graph according to an embodiment of the present disclosure.

[0010] FIG.5 illustrates an exemplary process for obtaining a representation of target data according to an embodiment of the present disclosure.

[0011] FIG.6 illustrates an exemplary process for deriving relation information.

[0012] FIG.7 illustrates an exemplary process for generating a data representation of target data based on an initial data representation and relation information of the target data according to an embodiment of the present disclosure.

[0013] FIG.8 illustrates an exemplary process for pre-training a representation obtaining model according to an embodiment of the present disclosure.

[0014] FIG.9 illustrates an exemplary process for classifying target data according to an embodiment of the present disclosure.

[0015] FIG.10 illustrates an exemplary process for calculating a matching degree between a query and semi-structured data according to an embodiment of the present disclosure.

[0016] FIG.11 is a flowchart of an exemplary method for obtaining a representation of target data according to an embodiment of the present disclosure.

[0017] FIG.12 illustrates an exemplary apparatus for obtaining a representation of target data according to an embodiment of the present disclosure.

[0018] FIG.13 illustrates an exemplary apparatus for obtaining a representation of target data according to an embodiment of the present disclosure.

DETAILED DESCRIPTION

[0019] The present disclosure will now be discussed with reference to several example implementations. It is to be appreciated that these implementations are discussed only for enabling those skilled in the art to better understand and thus implement the embodiments of the present disclosure, rather than suggesting any limitations on the scope of the present disclosure.

[0020] After receiving a user's query such as a question, a search engine may first obtain a representation of the query and a candidate answer with a trained language model, e.g., a Bidirectional Encoder Representations from Transformers (BERT) model. The candidate answer may be obtained from a database connected to or contained in the search engine, including, e.g., text paragraphs, semi-structured data, etc. Then, a matching degree between the query and the candidate answer may be calculated based on the obtained representation, and a candidate answer with a highest matching degree may be selected and presented to the user. At present, in the case that a candidate answer is semi-structured data, when obtaining a representation of the semi-structured data and a query, the semi-structured data is usually regarded as a flat document with multiple text units, and the semi-structured data is converted to a form suitable for language model processing through arranging these text units in order. This method does not take into account a structure of the semi-structured data at all. Alternatively, another method for obtaining a representation of semi-structured data and a query generates information related to a structure of the semi-structured data through encoding row and column values of each element in the semi-structured data, and then use this information to obtain the representation of the semi-structured data and the query. This method only implicitly considers the structure of the semi-structured data.

[0021] Embodiments of the present disclosure propose to obtain a representation of target data through effectively using an intrinsic structure or relations of the target data, and the process for obtaining the representation of the target data may also be considered as corresponding to a representation learning process of the target data. Herein, the representation may refer to a collection of information generated based on raw data and having a form that is conducive to machine learning model processing. The target data may refer to data at least a part of which does not fully meet representation and storage requirements of a relational database but contains tags for separating semantic elements

and layering fields, e.g., data including at least semi-structured data. Therefore, the core or foundation of the representation learning of the target data may be a representation learning of the semi-structured data. The target data may include only semi-structured data, or include semi-structured data and other information. For example, in a scenario where target data is classified, the target data may only include semi-structured data. For example, in the QA scenario, in a preliminary selection stage of preliminary selection of semi-structured data matching a query from candidate semi-structured data as candidate answers, only the candidate semi-structured data may be used as target data, while in the further selection stage of further selecting semi-structured data that more closely matches the query from the preliminary selected semi-structured data, a combination of the semi-structured data and the query may be used as the target data. Moreover, alternatively, the QA scenario may also include only one selection stage. Accordingly, in the selection stage of selecting semi-structured data matching a query from candidate semi-structured data as candidate answers, only the candidate semi-structured data may be used as the target data, or a combination of the candidate semi-structured data and the query may be used as the target data. The embodiments of the present disclosure propose to more effectively capture information related to a structure in target data through explicitly modeling components in the target data and relations among the components. Herein, the components may refer to parts constituting target data, and the relations among the components may refer to associations based on a predetermined criterion existing among the components. Since the structure of the target data can reflect semantic relations among its various elements, effective use of its structure when obtaining a representation of the target data can help to obtain a rich representation of the target data, thereby getting a more accurate result when using this representation to perform a further task. For example, in the scenario of classifying target data, that is, when the target data only includes semi-structured data, a rich representation of the target data can help to obtain a more accurate class prediction result for the semi-structured data; while in the QA scenario, that is, when the target data includes only semi-structured data, or includes both semi-structured data and a query, a rich representation of the target data can help to more accurately select semi-structured data that more closely matches the query.

[0022] In one aspect, the embodiments of the present disclosure propose to consider relations among a plurality of components in target data when obtaining a representation of the target data. For example, the representation of the target data may be obtained with relation information for representing the relations among the components. The relation

information may be derived from a graph corresponding to the target data, for example. In the graph, the components in the target data may be represented by nodes, and the relations among the components may be represented by edges among the nodes.

[0023] In another aspect, the embodiments of the present disclosure propose to, when
5 obtaining a representation of target data, aggregate representations of neighbor nodes of each node corresponding to each component of the target data with relation information, and update a representation of the node with the aggregated representations of the neighbor nodes, wherein updated representations of various nodes may be combined into a representation of the target data. Herein, a neighbor node of a node may refer to a node
10 with an edge between the node. In this way, the intrinsic semantic relations between neighbor nodes may be propagated through structural connections in the graph, so that information related to the structure in the target data can be better captured.

[0024] In yet another aspect, the embodiments of the present disclosure propose to obtain a representation of target data through a deep learning-based model, and propose
15 pre-training strategies to guide the model to be able to learn data representations from semantics embedded in both text and structure of the target data. Herein, a model used to obtain a representation of target data may be referred to as a representation obtaining model. In a pre-training strategy, all tokens corresponding to one or more components in a target data sample may be masked, and for each token, the token is predicted with a
20 contextual representation corresponding to the token, thereby training the representation obtaining model. This pre-training strategy can guide the representation obtaining model to better learn the contextual representation at the token level. In another pre-training strategy, alternatively or in addition, for each component of one or more components whose tokens are masked, a token corresponding to the component may be predicted with
25 representations of neighbor nodes of a node corresponding to the component, thereby training the representation obtaining model. This pre-training strategy can enable intrinsic semantic relations between neighbor nodes to be propagated via structural connections.

[0025] It should be appreciated that although the foregoing discussion and the following discussion may involve obtaining an example of obtaining a representation of
30 target data including a table, the embodiments of the present disclosure are not limited to this, but may obtain a representation of target data including other types of semi-structured data, such as a list, in a similar manner.

[0026] FIG.1 illustrates exemplary semi-structured data 100a and 100b. The semi-structured data 100a may be a table extracted from the Web, and the semi-structured data

100b may be a list extracted from the Web. Components in the semi-structured data may be systematically divided into multiple types, such as caption, content, header, cell, subject, attribute, etc.

[0027] The caption may refer to a component that summarizes information in the semi-structured data involved in the semi-structured data, and the component has a caption type. For example, a caption 102 of the semi-structured data 100a may be "Top 3 cities in GDP in the world", which indicates that the semi-structured data relates to cities sorted by GDP, and a caption 124 of the semi-structured data 100b may be "Five places suitable for skydiving", which indicates that the semi-structured data relates to places suitable for skydiving. Taking the QA scenario as an example, the terms in the caption of semi-structured data may be used to reliably determine a matching degree between a query and the semi-structured data.

[0028] The content may refer to a main data part of the semi-structured data, e.g., a part other than the caption in the semi-structured data which may be located, e.g., below the caption. For a table, the content may consist of multiple rows and multiple columns. For example, content 104 of the semi-structured data 100a may consist of four rows and three columns. For a list, the content may only have a single column. For example, content 126 of the semi-structured data 100b has only one column. Components in the content may be determined to have a content type.

[0029] For a table, components in the content that describe a data category may be referred to as headers, that is, these components have a header type, and components in the content other than the headers are referred to as cells, that is, these components have a cell type. A header may describe a category to which corresponding components, that is, cells, belong, and each cell may be considered as an instance of the category. In one case, the headers may be located in the first line of the content. In other cases, the headers may also be located in the first column of the content. It should be appreciated that, although the following discussion takes the headers being in the first line of the content as an example, the embodiments of the present disclosure may also be similarly applied to the case where the headers are in the first column of the content. For example, the semi-structured data 100a may have a header set 106 and a cell set 108. The header set 106 may include three headers "City", "GDP (100 million US dollars)" and "Country". Taking a header "Country" as an example, it may describe a category to which cells below it, such as "United States", "Japan", and "United States", belong, that is, country.

[0030] In the vertical direction, the cells of the table may also be organized into

multiple columns. For example, the cell set 108 may be organized into three columns, that is, a column 112, a column 114, and a column 116. The columns may include, e.g., a subject column, an attribute column, etc. The subject column may include one or more subjects or entities involved in the table, and the attribute column may include attribute information of a corresponding subject or entity. The subject column may be identified through, e.g., a heuristic method. For example, a proportion of different character strings contained in each column may be calculated, and a column with a highest proportion may be identified as the subject column. For example, the subject column of the semi-structured data 100a may be the column 112, and the attribute columns may be the column 114 and the column 116. Cells in the subject column are subject cells with a subject type, and cells in the attribute column are attribute cells with an attribute type. For example, in the semi-structured data 100a, a cell 118 "Tokyo" may be the subject cell, and a cell 120 "7590" and a cell 122 "Japan" may be the attribute cells corresponding to the subject "Tokyo".

[0031] For a list, the content of the list may not include a header, and all components in the content may be determined to have a cell type. For example, all components in the content 126 of the semi-structured data 100b may be referred to as cells, such as a cell 128 "3. Dubai - Palm Jumeirah", a cell 130 "4. United States - Snohomish", etc.

[0032] It should be appreciated that the component types of the semi-structured data described above in conjunction with FIG.1 are only exemplary. According to actual application requirements, the components of the semi-structured data may also have other types, or may include fewer types. For example, although both the semi-structured data 100a and 100b in FIG.1 include the caption, there may be no caption in some semi-structured data.

[0033] According to the embodiments of the present disclosure, relations among components of target data may be explicitly modeled. Modeling the relations can fuse semantic relations among the components and achieve a richer representation of the target data.

[0034] The target data may include only semi-structured data. For such target data, whether there is a relation between each two components in a plurality of components in the semi-structured data may be determined based at least on, e.g., types of the two components. Optionally, what kind of relation exists between the two components may also be determined.

[0035] In an embodiment, if types of two components are a caption and content,

respectively, there is a relation between the two components, and the relation is a caption-content relation. As described above, the content may include a header set and a cell set. According to the embodiments of the present disclosure, there is the caption-content relation between the caption and each component in the header set and the cell set. For example, for the semi-structured data 100a, there may be the caption-content relation between the caption 102 and the header 110, and there may be the caption-content relation between the caption 102 and the cell 118; and for the semi-structured data 100b, there may be the caption-content relation between the caption 124 and the cell 128.

[0036] In another embodiment, if types of two components are a header and a cell, and the two components are located in a same column, then there is a relation between the two components, and the relation is a header-cell relation. Because a header can describe a category to which corresponding cells belong, and each cell may be considered as an instance of the category, the header-cell relation may usually be a category-instance relation. For example, for the semi-structured data 100a, there may be the header-cell relation between the header 110 and the cell 122. Since there is no header in a list, there is no header-cell relation in the list.

[0037] In yet another embodiment, if types of two components are a subject and an attribute, respectively, that is, the two components are a subject cell and an attribute cell, respectively, and the two components are in a same row, then there is a relation between the two components, and the relation is a subject-attribute relation. Since the subject cell may usually contain a name of an entity, and the attribute cell may usually contain an attribute of the entity, the subject-attribute relation is usually an entity-attribute relation. For example, for the semi-structured data 100a, there may be the subject-attribute relation between the cell 118 and the cell 120. Since content of a list only has a single column, there is no subject-attribute relation in the list.

[0038] In still another embodiment, if types of both two components are a cell and the two components are located next to each other in a same column, then there is a relation between the two components, and the relation is a cell-cell relation. In addition, if types of both two components are an attribute, that is, the two components are attribute cells, and the two components are located next to each other in a same row, there is a cell-cell relation between the two components. Different cells in a same column may usually be different instances of a same category, and different attribute cells in a same row may usually contain different attributes of a same entity. For example, for the semi-structured data 100a, there may be the cell-cell relation between the cell 120 and the cell 122; and for

the semi-structured data 100b, there may be the cell-cell relation between the cell 128 and the cell 130.

[0039] It should be appreciated that the criteria for determining the relations among the components in the semi-structured data described above in conjunction with FIG.1 are only exemplary. According to actual application requirements, other criteria may also be used to determine the relations among the components in the semi-structured data.

[0040] In addition to semi-structured data, target data may also include other information, such as a query. Taking target data also includes a query as an example, the query may be regarded as one of components of the target data. The target data including both the semi-structured data and the query may have relations between the query and each component of the semi-structured data in addition to relations among the components in the semi-structured data. The relations between the query and each component of the semi-structured data may include, e.g., a query-caption relation, a query-content relation, etc. Herein, a query-caption relation may refer to a relation between a query and a caption of semi-structured data, and a query-content relation may refer to a relation between a query and each component in content of semi-structured data. For example, a query corresponding to the semi-structured data 100a may be "City with the highest GDP in the world", and a query corresponding to the semi-structured data 100b may be "Which place is best for skydiving?". There may be the query-caption relation between the query "City with the highest GDP in the world" and the caption 102 in the semi-structured data 100a, and there may be the query-content relation between the query and each component of the content 104. There may be the query-caption relation between the query "Which place is best for skydiving?" and the caption 124 in the semi-structured data 100b, and there may be the query-content relation between the query and each component of the content 126.

[0041] According to the embodiments of the present disclosure, a graph corresponding to target data may be constructed, and the graph may be used to derive relation information representing relations among a plurality of components in the target data. For example, the graph corresponding to the target data may be constructed based on components of the target data and the relations among the components. FIG.2 illustrates an exemplary process 200 for constructing a graph corresponding to target data according to an embodiment of the present disclosure. The process 200 may be applicable to target data including only semi-structured data, or may be applicable to target data including semi-structured data and other information.

[0042] At 210, a plurality of components in target data may be set as a plurality of

nodes.

[0043] At 220, for every two nodes in the plurality of nodes, it may be determined whether there is a relation between two components corresponding to the two nodes. Whether there is a relation between the two components may be determined based at least on, e.g., types of the two components, as described above in conjunction with FIG.1. For example, if one of the following conditions is met, it may be considered that there is a relation between the two components: the types of the two components are a caption and content, respectively; the types of the two components are a header and a cell, respectively, and the two components are located in a same column; the types of the two components are a subject cell and an attribute cell, and the two components are located in a same row; the types of both the two components are a cell, and the two components are located next to each other in a same column; or the types of both the two components are an attribute cell, and the two components are located next to each other in a same row.

[0044] At 230, in response to determining that there is a relation between the two components corresponding to the two nodes, it may be determined that there is an edge between the two nodes.

[0045] The steps 220 and 230 may be performed for every two nodes in the plurality of nodes. At 240, a set of edges among the plurality of nodes may be obtained.

[0046] At 250, the plurality of nodes and the obtained set of edges may be combined into a graph corresponding to the target data.

[0047] It should be appreciated that the process 200 in FIG.2 is only an example of a process for constructing a graph corresponding to target data. According to actual application requirements, the process for constructing the graph corresponding to the target data may include any other steps, and may include more or fewer steps. For example, in the case where semi-structured data in target data is a table, before the step 210 of setting a plurality of components of the target data as a plurality of nodes, optionally, whether the table is a vertical table may be detected first, and if it is detected that the table is the vertical table, the table is transposed to a horizontal table. Herein, a vertical table may refer to a table whose header is a vertical column, and a horizontal table may refer to a table whose header is a horizontal row. In addition, when it is determined that there is an edge between two nodes, optionally, a direction of the edge between the two nodes may be determined based on a relation type between two components corresponding to the two nodes, and arrows may be used to indicate the direction in the graph. In an embodiment, if a relation type between two components corresponding to two nodes is a header-cell

relation, a direction of an edge between the two nodes may be from a header node to a cell node. In another embodiment, if a relation type between two components corresponding to two nodes is a subject-attribute relation, a direction of an edge between the two nodes may be from a subject node to an attribute node. In yet another embodiment, if a relation type between two components corresponding to two nodes is one of a caption-content relation, a cell-cell relation, a query-caption relation, and a query-content relation, then a direction of an edge between the two nodes may be bidirectional. It should be appreciated that the aforementioned criteria for determining a direction of an edge between two nodes are only exemplary. According to actual application requirements, other criteria may also be used to determine a direction of an edge between two nodes.

[0048] FIG.3 illustrates exemplary graphs 300a and 300b according to an embodiment of the present disclosure. The graphs 300a and 300b may be graphs constructed through, e.g., the process 200 in FIG.2 and corresponding to the semi-structured data 100a and 100b in FIG.1, respectively.

[0049] In the graph 300a, a block C indicates a caption node $\mathcal{N}_C$ corresponding to the caption 102 in FIG.1, blocks $\{h_j\}$ indicate header nodes $\{\mathcal{N}_{h_j}\}$ corresponding to various headers in the header set 106 in FIG.1, and blocks $\{c_{ij}\}$ indicates cell nodes $\{\mathcal{N}_{c_{ij}}\}$ corresponding to various cells in the cell set 108 in FIG.1, where $2 \leq i \leq 4$ and $1 \leq j \leq 3$. A set of edges among these nodes are shown in the graph 300a. For example, there is an edge between the caption node $\mathcal{N}_C$ and each header node $\{\mathcal{N}_{h_j}\}$, and a direction of the edge is bidirectional, because a relation type between two components corresponding to the caption node $\mathcal{N}_C$ and the header node $\{\mathcal{N}_{h_j}\}$ is a caption-content relation; there is an edge between the caption node $\mathcal{N}_C$ and each cell node $\{\mathcal{N}_{c_{ij}}\}$, and the edge is bidirectional, because a relation type between two components corresponding to the caption node $\mathcal{N}_C$ and the cell node $\{\mathcal{N}_{c_{ij}}\}$ is a caption-content relation; there is an edge between the header node $\{\mathcal{N}_{h_j}\}$ and the cell node $\{\mathcal{N}_{c_{ij}}\}$ in the same column, and a direction of the edge is from the header node $\{\mathcal{N}_{h_j}\}$ to the cell node $\{\mathcal{N}_{c_{ij}}\}$, because a relation type between two components corresponding to the header node $\{h_j\}$ and the cell node $\{c_{ij}\}$ is a header-cell relation. The graph 300a also illustrates edges among cell nodes, edges between subject nodes and attribute nodes, etc.

[0050] In the graph 300b, a block C indicates a caption node $\mathcal{N}_C$ corresponding to the caption 124 in FIG.1, and blocks $\{l_i\}$ indicates cell nodes $\{\mathcal{N}_{l_i}\}$ corresponding to various

cells in the content 126 in FIG.1, where $1 \leq i \leq 5$. A set of edges among these nodes are shown in the graph 300b. For example, there is an edge between the caption node $\mathcal{N}_C$ and each cell node $\{\mathcal{N}_{l_i}\}$, and the edge is bidirectional, because a relation type between two components corresponding to the caption node $\mathcal{N}_C$ and the cell node $\{\mathcal{N}_{l_i}\}$ is a caption-content relation; and there is an edge between two adjacent cell nodes $\{\mathcal{N}_{l_i}\}$, and the edge is bidirectional, because a relation type between two components corresponding to the two adjacent cell nodes $\{\mathcal{N}_{l_i}\}$ is a cell-cell relation.

[0051] When target data includes semi-structured data and other information, a graph corresponding to the target data may also be constructed through the process 200 in FIG.2.

FIG.4 illustrates another exemplary graph 400 according to an embodiment of the present disclosure. The graph 400 may be a graph constructed through, e.g., the process 200 in FIG.2 and corresponding to target data including both a query and the semi-structured data 100a in FIG. 1. Compared with the graph 300a in FIG.3, the graph 400 also has a query node $\mathcal{N}_Q$ indicated by a block Q , which is used to represent a node corresponding to the query. In addition to the same edges as the set of edges shown in the graph 300a, edges associated with the query node $\mathcal{N}_Q$ are shown in the graph 400. For example, there is an edge between the query node $\mathcal{N}_Q$ and the caption node $\mathcal{N}_C$, and the edge is bidirectional, because a relation type between two components corresponding to the query node $\mathcal{N}_Q$ and the caption node $\mathcal{N}_C$ is a query-caption relation; there is an edge between the query node $\mathcal{N}_Q$ and the header node $\{\mathcal{N}_{h_j}\}$, and the edge is bidirectional, because a relation type between two components corresponding to the query node $\mathcal{N}_Q$ and the header node $\{\mathcal{N}_{h_j}\}$ is a query-content relation; and there is an edge between the query node $\mathcal{N}_Q$ and the cell node $\{\mathcal{N}_{c_{ij}}\}$, and the edge is bidirectional, because a relation type between two components corresponding to the query node $\mathcal{N}_Q$ and the cell node $\{\mathcal{N}_{c_{ij}}\}$ is a query-content relation.

[0052] It should be appreciated that the graphs 300a-300b and the graph 400 shown in FIGS. 3 and 4, respectively, are merely examples of graphs corresponding to target data. According to actual application requirements, the graph corresponding to the target data may have any other structure and may include other elements.

[0053] FIG.5 illustrates an exemplary process 500 for obtaining a representation of target data according to an embodiment of the present disclosure. For example, the representation of the target data may be obtained through a representation obtaining model

504. The representation obtaining model 504 may obtain the representation of the target data with relations among a plurality of components in the target data.

[0054] First, target data 502 may be obtained. The target data 502 may, e.g., include only semi-structured data, or include both semi-structured data and a query.

5 **[0055]** The target data 502 may be provided to a contextual language model 506 in the representation obtaining model 504. The contextual language model 506 may be, e.g., a transformer-based language model, such as a BERT model. The contextual language model 506 may generate a contextual representation of the target data 502, that is, an initial data representation 508. Herein, a representation generated by the contextual
10 language model may be referred to as an initial data representation.

[0056] The target data 502 may also be provided to a graph constructing module 510 in the representation obtaining model 504. The graph constructing module 510 may construct a graph 512 corresponding to the target data 502 through parsing a structure of the target data 502. For example, the graph constructing module 510 may construct the
15 graph 512 corresponding to the target data 502 through the process 200 in FIG.2.

[0057] Relation information 514 may be derived from the graph 512. The relation information 514 may be used to represent relations among a plurality of components in the target data 502. An exemplary process for deriving the relation information will be illustrated later in conjunction with FIG.6.

20 **[0058]** A graph reasoning module 516 in the representation obtaining model 504 may generate a data representation 518 of the target data 502 based on the initial data representation 508 and the relation information 514. Herein, a representation generated based on an initial data representation and relation information may be referred to as a data representation. Since the relation information used to generate the data representation of
25 the target data may be derived from the graph related to the target data, that is, information associated with the graph is taken into account when generating the data representation, the data representation may also be referred to as a graph representation. The graph reasoning module 516 may be, e.g., a two-layer Graph Convolutional Network (GCN). An exemplary process for generating the data representation based on the initial data
30 representation and the relation information will be illustrated later in conjunction with FIG.7.

[0059] It should be appreciated that the process 500 in FIG.5 is only an example of a process for obtaining a representation of target data. According to actual application requirements, the process for obtaining the representation of the target data may include

any other steps, and may include more or fewer steps. Moreover, the specific order or hierarchy of the steps in the process 500 is only exemplary, and the process for obtaining the representation may be performed in an order different from the described order.

[0060] In the following, a specific process for obtaining a data representation of target data is explained by taking the target data including both a query Q and semi-structured data S which is a table as an example. The target data may, e.g., correspond to the target data 502 in FIG.5.

[0061] The semi-structured data S may have a caption, and its content has M rows and N columns. Strings in the query Q and the semi-structured data S may be concatenated to obtain a string sequence $G = \{Q, C, \{h_j\}, \{c_{ij}\}\}$, where C represents the caption in the semi-structured data S , $\{h_j\}$ represents a header set in the semi-structured data S , $\{c_{ij}\}$ represents a cell set in the semi-structured data S , and $2 \leq i \leq M$ and $1 \leq j \leq N$. The string sequence G may be provided to a contextual language model, such as the contextual language model 506 in FIG.5. The contextual language model may generate a contextual representation of the string sequence G , and the contextual representation may be represented as, e.g., $\mathcal{LM}(G)$. The representation $\mathcal{LM}(G)$ may be an initial data representation of the target data, e.g., corresponding to the initial data representation 508 in FIG.5.

[0062] An initial node representation of a node corresponding to each component in the target data may be derived from the initial data representation $\mathcal{LM}(G)$. Herein, an initial node representation may refer to an initial representation of a node. Different nodes may contain tokens of different lengths. According to the embodiments of the present disclosure, a representation corresponding to a token of various lengths of each node may be converted into a fixed-size vector, and the vector may be used as an initial node representation of the node. In an implementation, a Bidirectional Long Short-Term Memory (BiLSTM) and a Multilayer Perceptron (MLP) may be applied on top of the contextual language model to convert the representation corresponding to the token of various lengths to the fixed-size vector. This process may be expressed as $MLP(\mathcal{B}(\mathcal{LM}(G))[s;t])$, where $\mathcal{B}(\mathcal{LM}(G))[s;t]$ represents a hidden state sequence corresponding to a token sequence from position s to position t output by the BiLSTM. Subscripts *start* and *end* may be used to indicate the start position and end position of a token of a corresponding node. Initial node representations of various nodes are shown in the following formulas:

$$\mathbf{Q} = \text{MLP}(\mathcal{B}(\mathcal{LM}(\mathbf{G}))[\mathbf{Q}_{start}; \mathbf{Q}_{end}]) \quad (1)$$

$$\mathbf{C} = \text{MLP}(\mathcal{B}(\mathcal{LM}(\mathbf{G}))[\mathbf{C}_{start}; \mathbf{C}_{end}]) \quad (2)$$

$$\mathbf{h}_j = \text{MLP}(\mathcal{B}(\mathcal{LM}(\mathbf{G}))[\mathbf{h}_{j, start}; \mathbf{h}_{j, end}]) \quad (3)$$

$$\mathbf{c}_{ij} = \text{MLP}(\mathcal{B}(\mathcal{LM}(\mathbf{G}))[\mathbf{c}_{ij, start}; \mathbf{c}_{ij, end}]) \quad (4)$$

5 where $\mathbf{Q}$ represents an initial node representation of the query Q , $\mathbf{C}$ represents an initial node representation of the caption C , $\mathbf{h}_j$ represents an initial node representation of the header h_j , and $\mathbf{c}_{ij}$ represents an initial node representation of the cell c_{ij} .

[0063] An initial data representation of the target data may be formed through combining the initial node representations of all nodes. The initial data representation may be expressed as $\mathbf{L}^{(0)} = \{\mathbf{Q}, \mathbf{C}, \{\mathbf{h}_j\}, \{\mathbf{c}_{ij}\}\} \in \mathbb{R}^{K \times d}$, where $K = 2 + M \times N$ is the same as a total number of nodes corresponding to components in the target data, and d is a output dimension of the MLP.

[0064] A graph corresponding to the target data may be constructed. For example, the graph corresponding to the target data may be constructed with the graph constructing module 510 in FIG.5 through the process 200 in FIG.2. Relation information for representing relations among a plurality of components in the target data may be derived from the graph corresponding to the target data. FIG.6 illustrates an exemplary process 600 for deriving relation information. Edge information between every two nodes in the graph may be acquired, a number of edges associated with each node in the graph may be counted, and the relation information may be derived based on the edge information and the number.

[0065] At 610, edge information between every two nodes may be acquired from a graph. Herein, edge information may refer to information used to express whether there is an edge between two nodes. Edge information between a node $\mathcal{N}_m$ and a node $\mathcal{N}_n$ may be expressed as, e.g., $A_{mn} = A_{nm} \in \{0,1\}$, where when there is an edge between the node $\mathcal{N}_m$ and the node $\mathcal{N}_n$, $A_{mn} = A_{nm} = 1$, and when there is no edge between the node $\mathcal{N}_m$ and the node $\mathcal{N}_n$, $A_{mn} = A_{nm} = 0$. Optionally, in the case that a direction of the edge is indicated in the graph, the edge information may be associated with the direction of the edge. For example, when the indicated direction of the edge in the graph is from the node $\mathcal{N}_m$ to the node $\mathcal{N}_n$, $A_{mn} = 1$ and $A_{nm} = 0$; while when the direction of the edge is bidirectional, $A_{mn} = A_{nm} = 1$.

[0066] The step 610 may be performed on every two nodes in the graph to obtain a set of edge information $\{A_{mn}\}$. At 620, the set of edge information may be combined into a

matrix $\mathbf{A} \in \mathbb{R}^{K \times K}$. The matrix $\mathbf{A}$ may be referred to as an adjacency matrix.

[0067] At 630, a number of edges associated with each node in the graph may be counted. For example, the number of edges associated with each node may be counted through summing the edge information associated with the node. For example, the number of edges associated with the node $\mathcal{N}_m$ may be expressed as $D_{mm} = \sum_n A_{mn}$.

[0068] The step 630 may be performed on each node in the graph to obtain a plurality of numbers of edges $\{D_{mm}\}$. At 640, the plurality of the number of edges may be combined into a matrix $\mathbf{D} \in \mathbb{R}^{K \times K}$. The matrix $\mathbf{D}$ may be referred to as a degree matrix. The degree matrix $\mathbf{D}$ may be, e.g., a diagonal matrix. In the diagonal matrix, all elements are 0 except the main diagonal.

[0069] At least the adjacency matrix $\mathbf{A}$ obtained at 620 and the degree matrix $\mathbf{D}$ obtained at 640 may be used as the relation information corresponding to the target data. The relation information may be further used to generate a data representation of the target data.

[0070] It should be appreciated that the process 600 in FIG.6 is only an example of a process for deriving relation information. According to actual application requirements, the process for deriving the relation information may include any other steps, and may include more or fewer steps. Moreover, the specific order or hierarchy of the steps in the process 600 is only exemplary, and the process for deriving the relation information may be performed in an order different from the described order.

[0071] FIG.7 illustrates an exemplary process 700 for generating a data representation of target data based on an initial data representation and relation information of the target data according to an embodiment of the present disclosure. For example, the initial data representation of the target data may include a plurality of initial node representations, and the plurality of initial node representations may respectively correspond to a plurality of components in the target data. The plurality of initial node representations may be updated to a plurality of node representations with the relation information, and the plurality of node representations may be combined into the data representation of the target data.

[0072] At 710, representations of neighbor nodes of a node corresponding to an initial node representation may be aggregated with relation information. As described above, the relation information may include at least e.g., an adjacency matrix $\mathbf{A}$ and a degree matrix $\mathbf{D}$.

[0073] At 720, the initial node representation may be updated to a node representation

based on the aggregated representations of neighbor nodes.

[0074] The steps 710 and 720 may be performed for initial node representations corresponding to various components. At 730, a plurality of node representations respectively corresponding to a plurality of components may be obtained.

5 [0075] At 740, the plurality of node representations may be combined into a data representation of the target data.

[0076] The process 700 may be performed through, e.g., the graph reasoning module 516 in FIG.5. The graph reasoning module 516 may be, e.g., a two-layer GCN. At each layer, GCN may aggregate the representations of neighbor nodes of each node of the target data, and update the representation of the node with the aggregated representations of neighbor nodes. As described above, the initial data representation of the target data may be expressed as, e.g., $\mathbf{L}^{(0)} = \{\mathbf{Q}, \mathbf{C}, \{\mathbf{h}_j\}, \{\mathbf{c}_{ij}\}\} \in \mathbb{R}^{K \times d}$. The process for generating the data representation based on the initial data representation and the relation information may be expressed by the following formula:

$$15 \quad \mathbf{L}^{(l)} = \sigma(\tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{L}^{(l-1)} \mathbf{W}^{(l-1)}) \quad (5)$$

where $\mathbf{L}^{(l)}$ represents the output of the l -th layer of GCN ($l = 1, 2$), σ is a non-linear activation function, such as a Rectified Linear Unit (ReLU), and $\mathbf{W}^{(l-1)}$ is a learnable weight matrix of the $(l-1)$ -th layer. In formula (5), the symbol “ $\sim$ ” indicates a renormalization operation, such as adding a self-connection to each node in the graph when constructing the adjacency matrix $\mathbf{A}$ or the degree matrix $\mathbf{D}$. This operation allows to update a representation of the l -th layer of each node with a representation of the $(l-1)$ -th layer of the node. $\mathbf{L}^{(2)}$ represents an updated representation obtained after two rounds of convolution, which may be used as a data representation.

[0077] In the process 700, intrinsic semantic relations between neighbor nodes may be propagated through structural connections in the graph, so that information related to the structure in the target data can be better captured. It should be appreciated that the process 700 is merely an example of a process of generating a data representation based on an initial data representation and relation information. According to actual application requirements, the process for generating the data representation may include any other steps, and may include more or fewer steps.

[0078] As described above, a data representation of target data may be obtained through a representation obtaining model. The embodiments of the present disclosure propose pre-training strategies that may guide the representation obtaining model to be

able to learn data representations from semantics embedded in both text and structure of target data. A first pre-training strategy may mask all tokens corresponding to one or more components in a target data sample, and for each token, predict the token with a contextual representation corresponding to the token, thereby training the representation obtaining model. The first pre-training strategy may also be referred to as a Whole Cell Masking (WCM) strategy. Alternatively or additionally, for each component of the one or more components whose tokens are masked, a second pre-training strategy may predict a token corresponding to the component with representations of neighbor nodes of a node corresponding to the component, thereby training the representation obtaining model. The second pre-training strategy may also be referred to as a Neighbor Prediction Objective (NPO) strategy. FIG.8 illustrates an exemplary process 800 for pre-training a representation obtaining model according to an embodiment of the present disclosure.

[0079] At 802, a target data sample may be obtained.

[0080] At 804, the target data sample may be converted into a token sequence. The token sequence may be expressed as $T = (t_1, \dots, t_{|X|})$, where $t_m \in T$ is the m -th token.

[0081] At 806, all tokens corresponding to one or more components in the token sequence T may be masked. According to an embodiment of the present disclosure, 15% of the total components may be randomly selected for masking. For each component in the selected components, all tokens corresponding to the component are masked. By way of example, in the masked tokens, 80% of the tokens are replaced with a special masked token [MASK], 10% of the tokens are replaced with random tokens, and 10% of the tokens remain as original tokens.

[0082] At 808, an initial data representation of the masked token sequence may be generated. The initial data representation may be generated through, e.g., a contextual language model, such as a BERT model. The initial data representation may be a contextual representation.

[0083] At 810, a contextual representation corresponding to a masked token may be extracted from the initial data representation. Taking a token t_m as an example, a contextual representation $\mathbf{e}_m \in \mathbb{R}^d$ corresponding to the token t_m may be extracted from the initial data representation.

[0084] At 812, a prediction result $\mathbf{t}_m \in \mathbb{R}^V$ corresponding to the token t_m may be obtained through decoding $\mathbf{e}_m$, where V represents a vocabulary size. In an implementation, $\mathbf{e}_m$ may be decoded through a MLP, as shown in the following formula:

$$\mathbf{t}_m = \text{MLP}(\mathbf{e}_m) \quad (6)$$

[0085] At 814, a first pre-training loss $\mathcal{L}_{WCM}(\mathbf{t}_m)$ may be calculated based on the token $\mathbf{t}_m$ and the prediction result $\mathbf{t}_m$. In an implementation, the first pre-training loss may be calculated through a cross-entropy function, as shown in the following formula:

$$\mathcal{L}_{WCM}(\mathbf{t}_m) = -\log P(\mathbf{t}_m | \mathbf{t}_m) \quad (7)$$

[0086] After the initial data representation of the token sequence is generated at 808, at 816, a data representation of the token sequence may be generated. The data representation may be generated based on e.g., the initial data representation and relation information of the target data sample through the process 700 in FIG. 7.

10 [0087] At 818, representations of one or more neighbor nodes of a node corresponding to the masked component may be acquired from the data representation of the token sequence. The representations of neighbor nodes may be obtained, e.g., through a function $\text{neighbour}(\cdot)$. Taking a node $\mathcal{N}_n$ as an example, representations of neighbor nodes of the node may be, e.g., $\text{neighbour}(\mathbf{L}_n^{(2)})$, where $\mathbf{L}_n^{(2)}$ is a node representation of the node $\mathcal{N}_n$.

15 [0088] At 820, a dimensionality reduction operation may be performed on the acquired representations of one or more neighbor nodes. In an implementation, the representations of one or more neighbor nodes may be converted into a d -dimensional vector with mean pooling. This operation may be expressed as, e.g., $\text{Pooling}(\text{neighbour}(\mathbf{L}_n^{(2)}))$.

20 [0089] At 822, a representation of the k -th token of a node $\mathcal{N}_n$, $\mathbf{r}_{nk}$, may be generated based on the representations of neighbor nodes and a positional representation of position k , $\mathbf{p}_k \in \mathbb{R}^d$, and the k -th token of the node $\mathcal{N}_n$ may correspond to the token $\mathbf{t}_m$ in the token sequence T . In an implementation, the representation of the k -th token of the node $\mathcal{N}_n$, $\mathbf{r}_{nk}$, may be generated through cascading the converted representations of one or
25 more neighbor nodes and the positional representation $\mathbf{p}_k$, as shown in the following formula:

$$\mathbf{r}_{nk} = [\text{Pooling}(\text{neighbour}(\mathbf{L}_n^{(2)})); \mathbf{p}_k] \quad (8)$$

[0090] At 824, a prediction result $\mathbf{t}_{nk} \in \mathbb{R}^V$ may be obtained through decoding $\mathbf{r}_{nk}$. In an implementation, $\mathbf{r}_{nk}$ may be decoded through a MLP with two hidden layers, as shown
30 in the following formula:

$$\mathbf{t}_{nk} = \text{MLP}(\mathbf{r}_{nk}) \quad (9)$$

[0091] At 826, a second pre-training loss $\mathcal{L}_{NPO}(\mathbf{t}_m)$ may be calculated based on the

token t_m and the prediction result t_{nk} . In an implementation, the second pre-training loss may be calculated through a cross-entropy function, as shown in the following formula:

$$\mathcal{L}_{NPO}(t_m) = -\log P(t_m|t_{nk}) \quad (10)$$

[0092] After the first pre-training loss and the second pre-training loss are obtained, at 5 828, a total pre-training loss $\mathcal{L}(t_m)$ may be calculated based on the first pre-training loss and the second pre-training loss. In an implementation, the total pre-training loss may be calculated as the sum of the first pre-training loss and the second pre-training loss, as shown in the following formula:

$$\mathcal{L}(t_m) = \mathcal{L}_{WCM}(t_m) + \mathcal{L}_{NPO}(t_m) = -\log P(t_m|t_m) - \log P(t_m|t_{nk}) \quad (11)$$

10 [0093] At 830, the representation obtaining model may be pre-trained through minimizing the total pre-training loss $\mathcal{L}(t_m)$.

[0094] The process 800 in FIG.8 may include both a Whole Cell Masking strategy and a Neighbor Prediction Objective strategy, wherein the steps 802-814 may correspond to the Whole Cell Masking strategy, and the steps 802-808 and steps 816-826 may correspond to the Neighbor Prediction Objective strategy. The Whole Cell Masking strategy may be used to pre-train to a contextual language model in a representation obtaining model, such as the contextual language model 506 in FIG.5, to optimize parameters of the contextual language model. The Whole Cell Masking strategy can guide the contextual language model to better learn contextual representations at token level.

20 The Neighbor Prediction Objective strategy may be used at least for pre-training a graph reasoning module in the representation obtaining model, such as the graph reasoning module 516 in FIG.5, to optimize parameters of the graph reasoning module. In addition, because the Neighbor Prediction Objective strategy uses initial data representations from the contextual language model, the Neighbor Prediction Objective strategy can also optimize the parameters of the contextual language model. The Neighbor Prediction Objective strategy can enable intrinsic semantic relations between neighbor nodes to be propagated through structural connections. It should be appreciated that the Whole Cell Masking strategy and the Neighbor Prediction Objective strategy may be applied separately, e.g., only the Whole Cell Masking strategy or only the Neighbor Prediction Objective strategy may be applied. In the case that only one pre-training strategy is used to pre-train the representation obtaining model, the representation obtaining model may be pre-trained through minimizing the pre-training loss corresponding to the pre-training strategy.

[0095] A data representation of target data obtained according to the embodiments of the present disclosure may be applied to various tasks related to the target data. For example, when the target data includes only semi-structured data, task such as a classification task, etc., may be performed on the target data with a data representation of the target data obtained according to the embodiments of the present disclosure. Taking the classification task as an example, some classes of target data may be predefined, such as target data related to cities, target data related to celebrities, etc., and then an input target data may be classified. FIG.9 illustrates an exemplary process 900 for classifying target data according to an embodiment of the present disclosure.

[0096] First, target data 902 may be obtained. The target data 902 may include only semi-structured data 904.

[0097] A data representation 908 of the target data 902 may be obtained through a representation obtaining model 906. The representation obtaining model 906 may correspond to, e.g., the representation obtaining model 504 in FIG.5.

[0098] The data representation 908 may be provided to a classifying module 910. The classifying module 910 may determine a class prediction result corresponding to the target data 902 based on the data representation 908. In an implementation, a class set related to the target data may be determined first. For each class in the class set, the classifying module 910 may calculate a predicted probability of marking the target data 902 as the class based on the data representation 908, and select a class whose predicted probability is the largest as the class prediction result.

[0099] Since a structure of the target data is effectively used when obtaining the data representation 908 of the target data 902, the obtained data representation 908 may be richer. The rich data representation can help to obtain a more accurate class prediction result of the semi-structured data 904.

[00100] It should be appreciated that the process 900 in FIG.9 is only an example of using a data representation of target data to perform a task when the target data only includes semi-structured data. According to actual application requirements, the data representation of the target data may also be used to perform other tasks. For example, in a QA scenario, assuming that target data only includes candidate semi-structured data, a data representation of the candidate semi-structured data may be obtained according to the embodiments of the present disclosure, and then a matching degree between a query and the candidate semi-structured data may be calculated based on the data representation of the candidate semi-structured data and an additionally obtained representation of the query.

In addition, a data representation of semi-structured data may be used to perform some other operations on the semi-structured data, such as summing, averaging, etc.

[00101] In addition, when target data includes semi-structured data and other information, some tasks may be performed on the target data through using a data representation of the target data obtained according to the embodiments of the present disclosure. Taking target data including semi-structured data and a query as an example, a matching degree between the query and the semi-structured data may be calculated through using a data representation of the target data obtained according to the embodiments of the present disclosure. FIG.10 illustrates an exemplary process 1000 for calculating a matching degree between a query and semi-structured data according to an embodiment of the present disclosure.

[00102] First, target data 1002 may be obtained. The target data 1002 may include both a query 1004 and semi-structured data 1006.

[00103] A data representation 1010 of the target data 1002 may be obtained through a representation obtaining model 1008. The representation obtaining model 1008 may correspond to, e.g., the representation obtaining model 504 in FIG.5.

[00104] The data representation 1010 may be provided to a scoring module 1012. The scoring module 1012 may calculate a matching degree between the query 1004 and the semi-structured data 1006 based on the data representation 1010. In an implementation, the matching degree between the query and the semi-structured data $d(Q, S)$ may be calculated through performing a mean pooling operation on the data representation 1010 first, and then decoding the mean pooled data representation with a MLP, as shown in the following formula:

$$d(Q, S) = MLP(Pooling(L^{(2)})) \quad (12)$$

where Q may represent the query 1004, S may represent the semi-structured data 1006, and $L^{(2)}$ may represent the data representation 1010.

[00105] Since a structure of the target data 1002, especially a structure of the semi-structured data 1006 in the target data 1002, is effectively used when obtaining the data representation 1010 of the target data 1002, the obtained data representation 1010 may be richer. The rich data representation can help to more accurately select the semi-structured data 1006 that more closely matches the query 1004.

[00106] It should be appreciated that the process 1000 in FIG.10 is only an example of using a data representation of target data to perform a task when the target data includes

semi-structured data and other information. According to actual application requirements, the data representation of the target data may also be used to perform other tasks. For example, depending on specific content of the other information included in the target data, the data representation of the target data may be used to perform tasks corresponding to the other information.

[00107] FIG.11 is a flowchart of an exemplary method 1100 for obtaining a representation of target data according to an embodiment of the present disclosure.

[00108] At 1110, target data may be obtained, the target data including at least semi-structured data.

[00109] At 1120, an initial data representation of the target data may be generated.

[00110] At 1130, relations among a plurality of components in the target data may be identified

[00111] At 1140, a data representation of the target data may be generated based on the initial data representation and the relations.

[00112] In an implementation, the relations may include at least one of a caption-content relation, a header-cell relation, a subject-attribute relation, and a cell-cell relation.

[00113] In an implementation, the method 1100 may further comprise: generating relation information representing the relations. The generating a data representation may comprise: generating the data representation based on the initial data representation and the relation information.

[00114] The generating relation information may comprise: constructing a graph corresponding to the target data; and deriving the relation information from the graph.

[00115] The constructing a graph may comprise: setting a plurality of components in the target data as a plurality of nodes; determining a set of edges among the plurality of nodes; and combining the plurality of nodes and the set of edges into the graph.

[00116] The determining a set of edges may comprise, for every two nodes in the plurality of nodes: determining whether there is a relation between two components corresponding to the two nodes; and in response to determining that there is a relation between the two components, determining that there is an edge between the two nodes.

[00117] The deriving the relation information may comprise: acquiring edge information between every two nodes in the graph; counting a number of edges associated with each node in the graph; and deriving the relation information based on the edge information and the number.

[00118] The initial data representation may comprise a plurality of initial node

representations, the plurality of initial node representations respectively corresponding to a plurality of components in the target data. The generating the data representation may comprise: updating the plurality of initial node representations to a plurality of node representations with the relation information; and combining the plurality of node representations into the data representation.

[00119] The updating the plurality of initial node representations may comprise, for each initial node representation: aggregating representations of neighbor nodes of a node corresponding to the initial node representation; and updating the initial node representation based on the aggregated representations of the neighbor nodes.

[00120] In an implementation, the initial data representation may be generated through a contextual language model. Pre-training of the contextual language model may comprise at least: masking all tokens corresponding to one or more components in a target data sample; and for each token in said all tokens, predicting the token with a contextual representation corresponding to the token.

[00121] In an implementation, the data representation may be generated through a graph reasoning module. Pre-training of the graph reasoning module may comprise at least: masking all tokens corresponding to one or more components in a target data sample; and for each component of the one or more components, predicting tokens corresponding to the component with representations of neighbor nodes of a node corresponding to the component.

[00122] In an implementation, the target data may further include a query.

[00123] The relations may include relations between the query and each component in the semi-structured data.

[00124] The method 1100 may further comprise: calculating a matching degree between the query and the semi-structured data based on the data representation.

[00125] In an implementation, the semi-structured data may include at least one of a table and a list.

[00126] It should be appreciated that the method 1100 may further comprise any steps/processes for obtaining a representation of target data according to the embodiments of the present disclosure as mentioned above.

[00127] FIG.12 illustrates an exemplary apparatus 2600 for obtaining a representation of target data according to an embodiment of the present disclosure.

[00128] The apparatus 1200 may comprise: a data obtaining module 1210, for obtaining target data, the target data including at least semi-structured data; an initial data

representation generating module 1220, for generating an initial data representation of the target data; a relation identifying module 1230, for identifying relations among a plurality of components in the target data; and a data representation generating module 1240, for generating a data representation of the target data based on the initial data representation and the relations.

[00129] In an implementation, the apparatus 1200 may further comprise a relation information generating module, for generating relation information representing the relations. The data representation generating module 1240 may be further configured for: generating the data representation based on the initial data representation and the relation information.

[00130] The relation information generating module may be further configured for: constructing a graph corresponding to the target data; and deriving the relation information from the graph.

[00131] The initial data representation may comprise a plurality of initial node representations, the plurality of initial node representations respectively corresponding to a plurality of components in the target data. The generating the data representation may comprise: updating the plurality of initial node representations to a plurality of node representations with the relation information; and combining the plurality of node representations into the data representation.

[00132] It should be appreciated that the apparatus 1200 may further comprise any other modules configured for obtaining a representation of target data according to the embodiments of the present disclosure as mentioned above.

[00133] FIG.13 illustrates an exemplary apparatus 1300 for obtaining a representation of target data according to an embodiment of the present disclosure.

[00134] The apparatus 1300 may comprise at least one processor 1310. The apparatus 1300 may further comprise a memory 1320 connecting with the processor 1310. The memory 1320 may store computer-executable instructions that, when executed, cause the processor 1310 to perform any operations of the methods for obtaining a representation of target data according to the embodiments of the present disclosure as mentioned above.

[00135] The embodiments of the present disclosure may be embodied in a non-transitory computer-readable medium. The non-transitory computer-readable medium may comprise instructions that, when executed, cause one or more processors to perform any operations of the methods for obtaining a representation of target data according to the embodiments of the present disclosure as mentioned above.

[00136] It should be appreciated that all the operations in the methods described above are merely exemplary, and the present disclosure is not limited to any operations in the methods or sequence orders of these operations, and should cover all other equivalents under the same or similar concepts.

5 **[00137]** It should also be appreciated that all the modules in the apparatuses described above may be implemented in various approaches. These modules may be implemented as hardware, software, or a combination thereof. Moreover, any of these modules may be further functionally divided into sub-modules or combined together.

[00138] Processors have been described in connection with various apparatuses and
10 methods. These processors may be implemented using electronic hardware, computer software, or any combination thereof. Whether such processors are implemented as hardware or software will depend upon the particular application and overall design constraints imposed on the system. By way of example, a processor, any portion of a processor, or any combination of processors presented in the present disclosure may be
15 implemented with a microprocessor, microcontroller, digital signal processor (DSP), a field-programmable gate array (FPGA), a programmable logic device (PLD), a state machine, gated logic, discrete hardware circuits, and other suitable processing components configured to perform the various functions described throughout the present disclosure. The functionality of a processor, any portion of a processor, or any combination of
20 processors presented in the present disclosure may be implemented with software being executed by a microprocessor, microcontroller, DSP, or other suitable platform.

[00139] Software shall be construed broadly to mean instructions, instruction sets, code, code segments, program code, programs, subprograms, software modules, applications, software applications, software packages, routines, subroutines, objects, threads of
25 execution, procedures, functions, etc. The software may reside on a computer-readable medium. A computer-readable medium may include, by way of example, memory such as a magnetic storage device (e.g., hard disk, floppy disk, magnetic strip), an optical disk, a smart card, a flash memory device, random access memory (RAM), read only memory (ROM), programmable ROM (PROM), erasable PROM (EPROM), electrically erasable
30 PROM (EEPROM), a register, or a removable disk. Although memory is shown separate from the processors in the various aspects presented throughout the present disclosure, the memory may be internal to the processors, e.g., cache or register.

[00140] The previous description is provided to enable any person skilled in the art to practice the various aspects described herein. Various modifications to these aspects will

be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other aspects. Thus, the claims are not intended to be limited to the aspects shown herein. All structural and functional equivalents to the elements of the various aspects described throughout the present disclosure that are known or later come
5 to be known to those of ordinary skilled in the art are intended to be encompassed by the claims.

CLAIMS

1. A method for obtaining a representation of target data, comprising:
obtaining target data, the target data including at least semi-structured data;
generating an initial data representation of the target data;
identifying relations among a plurality of components in the target data; and
generating a data representation of the target data based on the initial data representation and the relations.
2. The method of claim 1, wherein the relations include at least one of a caption-content relation, a header-cell relation, a subject-attribute relation, and a cell-cell relation.
3. The method of claim 1, further comprising:
generating relation information representing the relations, and
wherein the generating a data representation comprises:
generating the data representation based on the initial data representation and the relation information.
4. The method of claim 3, wherein the generating relation information comprises:
constructing a graph corresponding to the target data; and
deriving the relation information from the graph.
5. The method of claim 4, wherein the constructing a graph comprises:
setting a plurality of components in the target data as a plurality of nodes;
determining a set of edges among the plurality of nodes; and
combining the plurality of nodes and the set of edges into the graph.
6. The method of claim 5, wherein the determining a set of edges comprises, for every two nodes in the plurality of nodes:
determining whether there is a relation between two components corresponding to the two nodes; and
in response to determining that there is a relation between the two components, determining that there is an edge between the two nodes.
7. The method of claim 4, wherein the deriving the relation information comprises:
acquiring edge information between every two nodes in the graph;
counting a number of edges associated with each node in the graph; and
deriving the relation information based on the edge information and the number.
8. The method of claim 3, wherein the initial data representation comprises a plurality of initial node representations, the plurality of initial node representations respectively corresponding to a plurality of components in the target data, and the

generating the data representation comprises:

updating the plurality of initial node representations to a plurality of node representations with the relation information; and

combining the plurality of node representations into the data representation.

9. The method of claim 8, wherein the updating the plurality of initial node representations comprises, for each initial node representation:

aggregating representations of neighbor nodes of a node corresponding to the initial node representation; and

updating the initial node representation based on the aggregated representations of the neighbor nodes.

10. The method of claim 1, wherein the initial data representation is generated through a contextual language model, and pre-training of the contextual language model comprises at least:

masking all tokens corresponding to one or more components in a target data sample; and

for each token in said all tokens, predicting the token with a contextual representation corresponding to the token.

11. The method of claim 1, wherein the data representation is generated through a graph reasoning module, and pre-training of the graph reasoning module comprises at least:

masking all tokens corresponding to one or more components in a target data sample; and

for each component of the one or more components, predicting tokens corresponding to the component with representations of neighbor nodes of a node corresponding to the component.

12. The method of claim 1, wherein the target data further includes a query.

13. The method of claim 12, wherein the relations include relations between the query and each component in the semi-structured data.

14. An apparatus for obtaining a representation of target data, comprising:

a data obtaining module, for obtaining target data, the target data including at least semi-structured data;

an initial data representation generating module, for generating an initial data representation of the target data;

a relation identifying module, for identifying relations among a plurality of components in the target data; and

a data representation generating module, for generating a data representation of the target data based on the initial data representation and the relations.

15. An apparatus for obtaining a representation of target data, comprising:

at least one processor; and

a memory storing computer-executable instructions that, when executed, cause the at least one processor to:

obtain target data, the target data including at least semi-structured data,

generate an initial data representation of the target data,

identify relations among a plurality of components in the target data, and

generate a data representation of the target data based on the initial data representation and the relations.

1/9

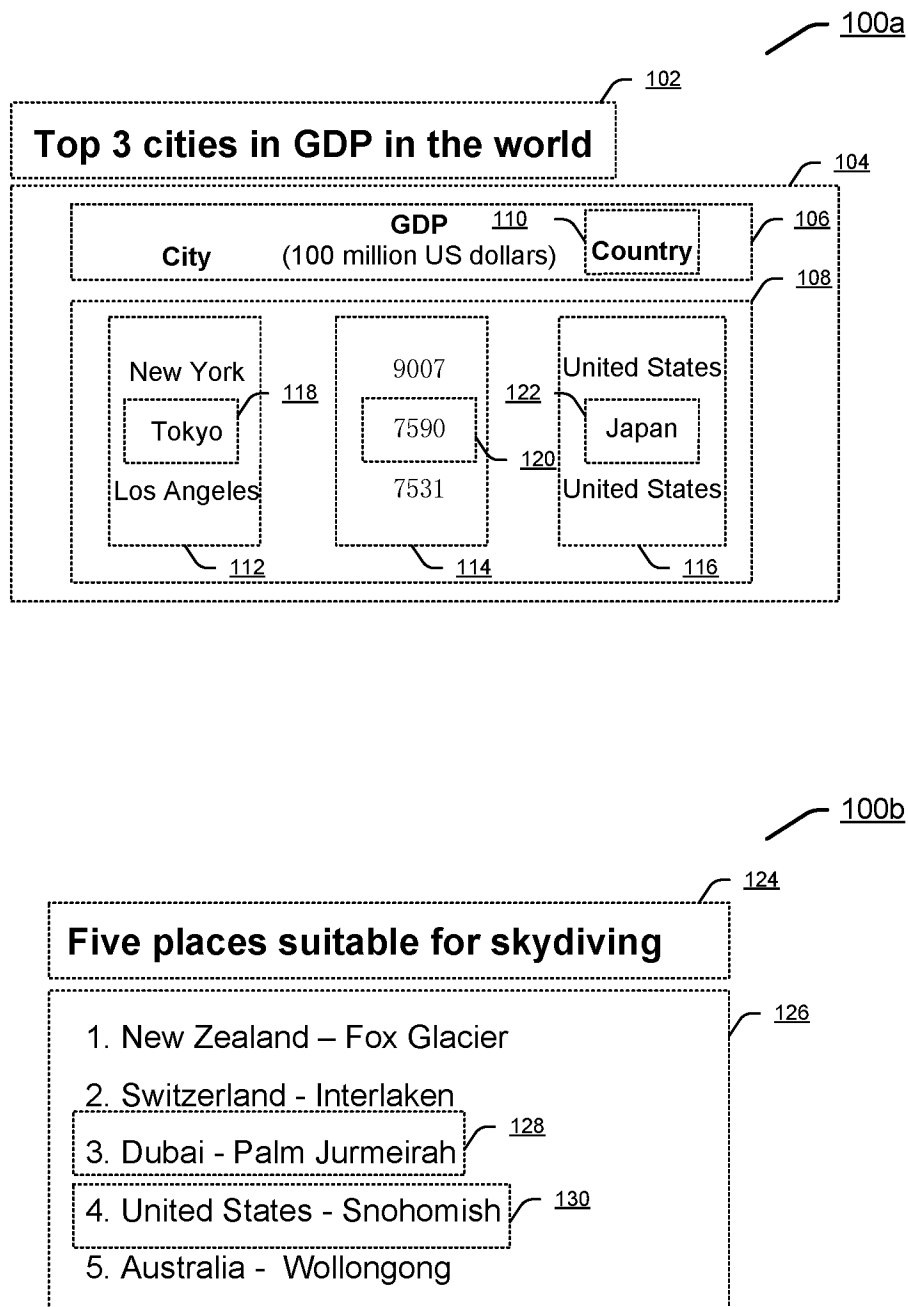


FIG 1

2/9

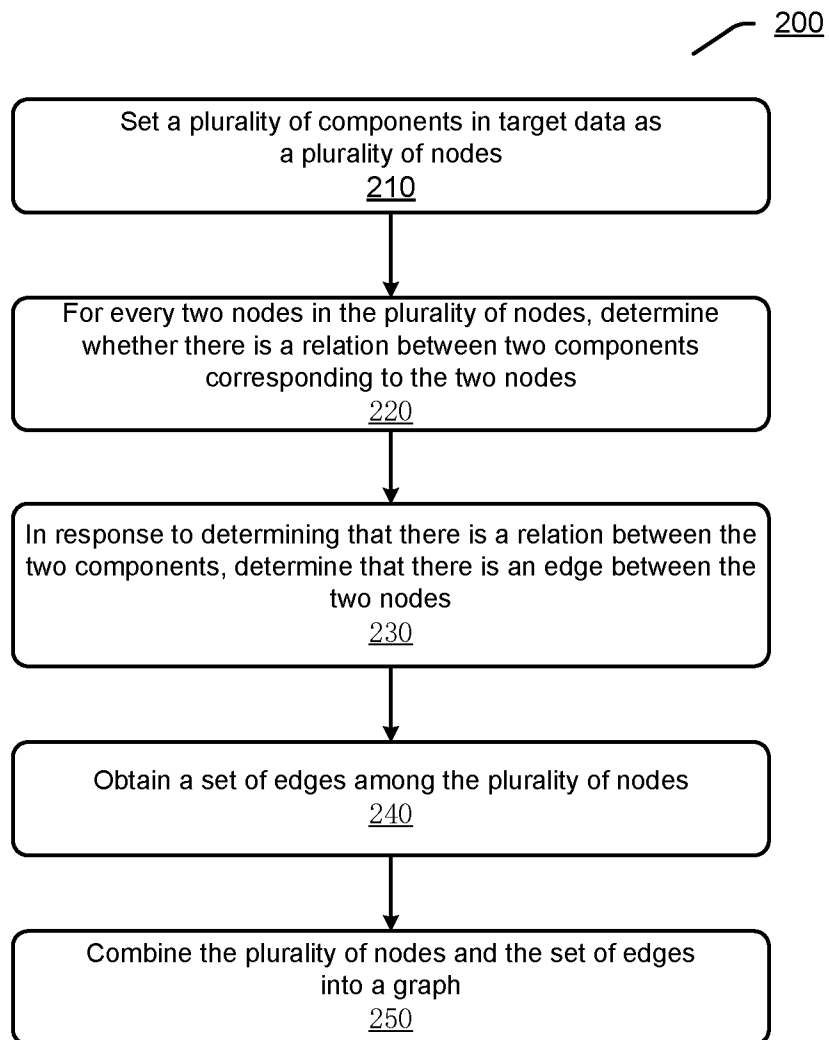


FIG 2

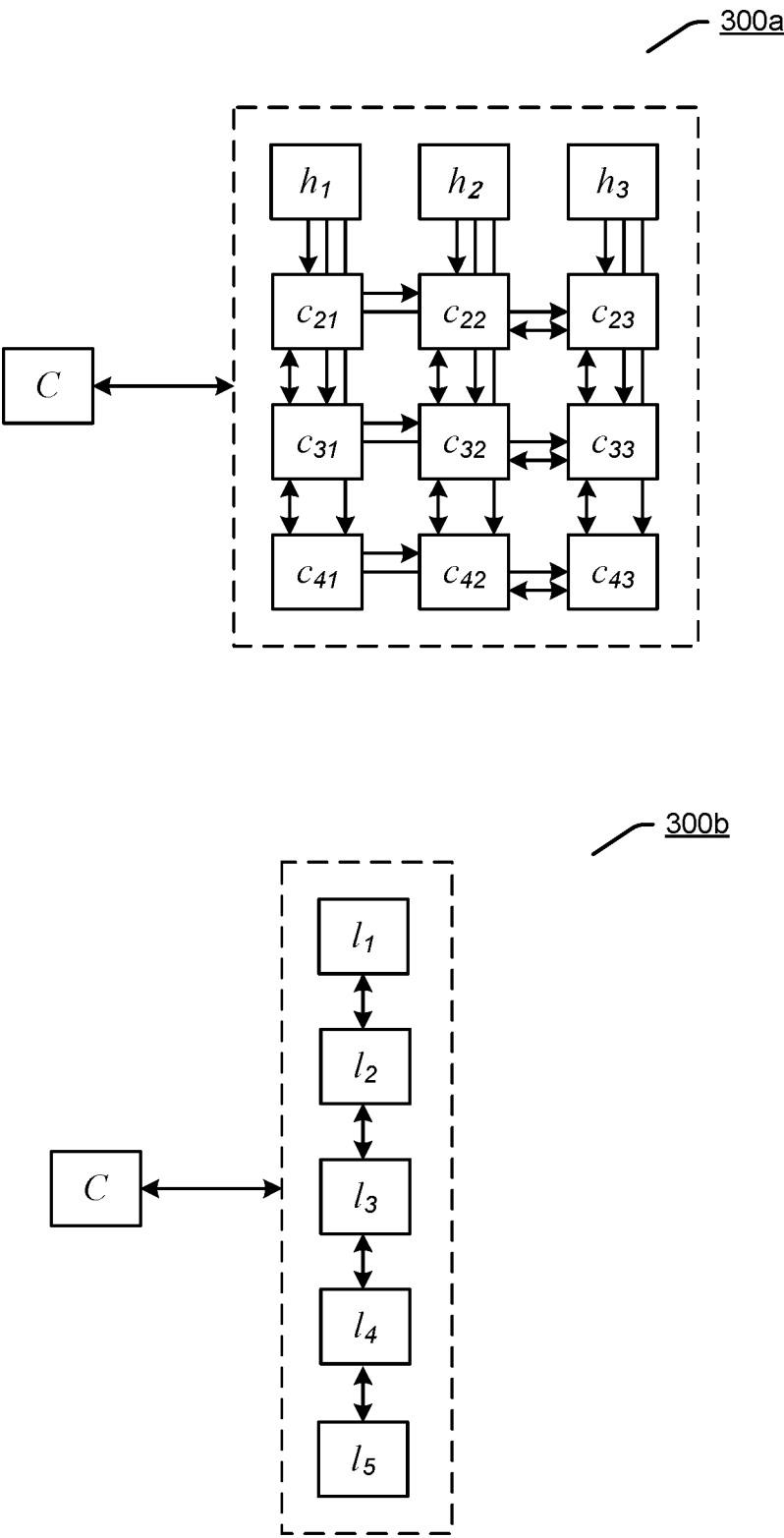


FIG 3

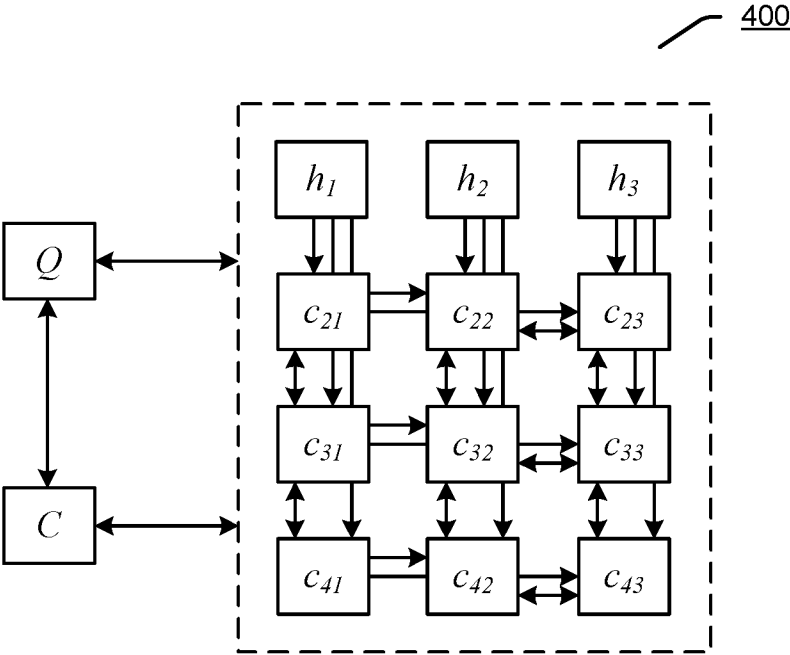


FIG 4

5/9

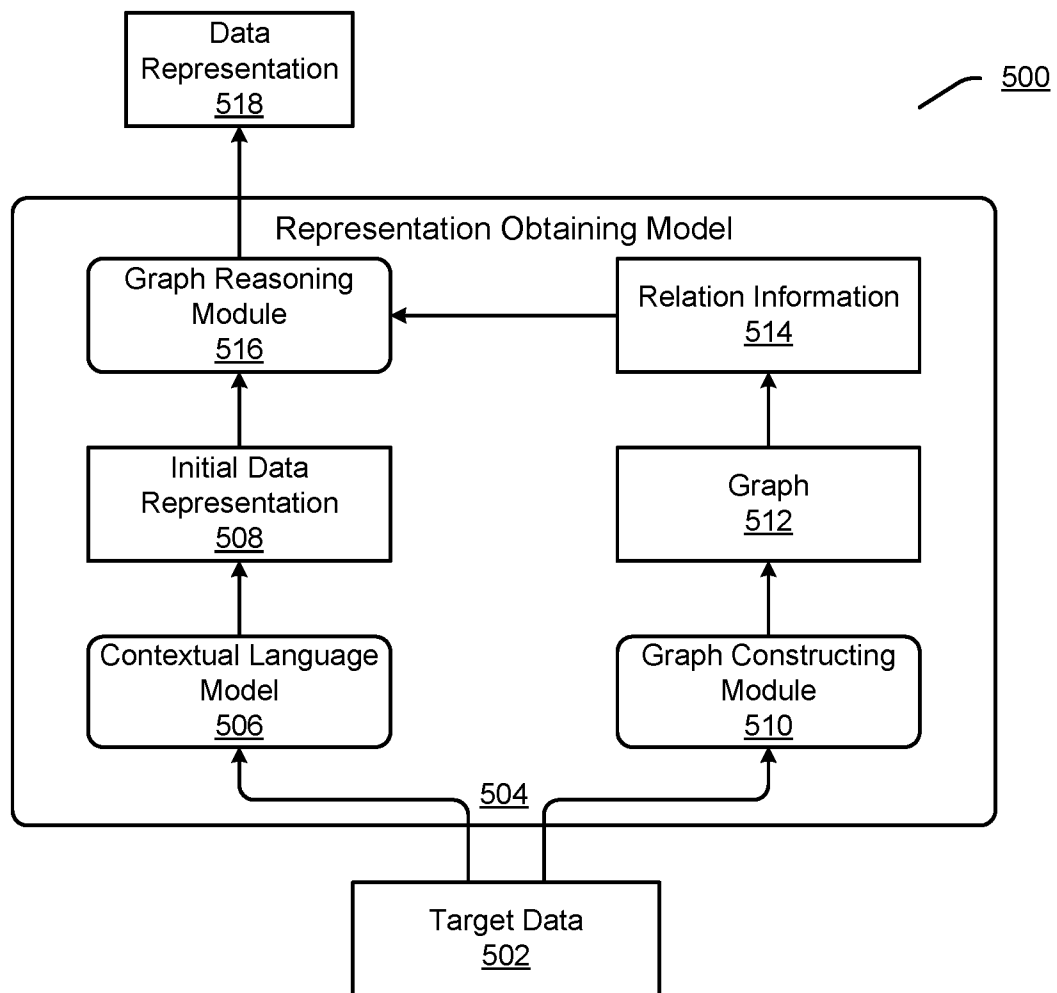
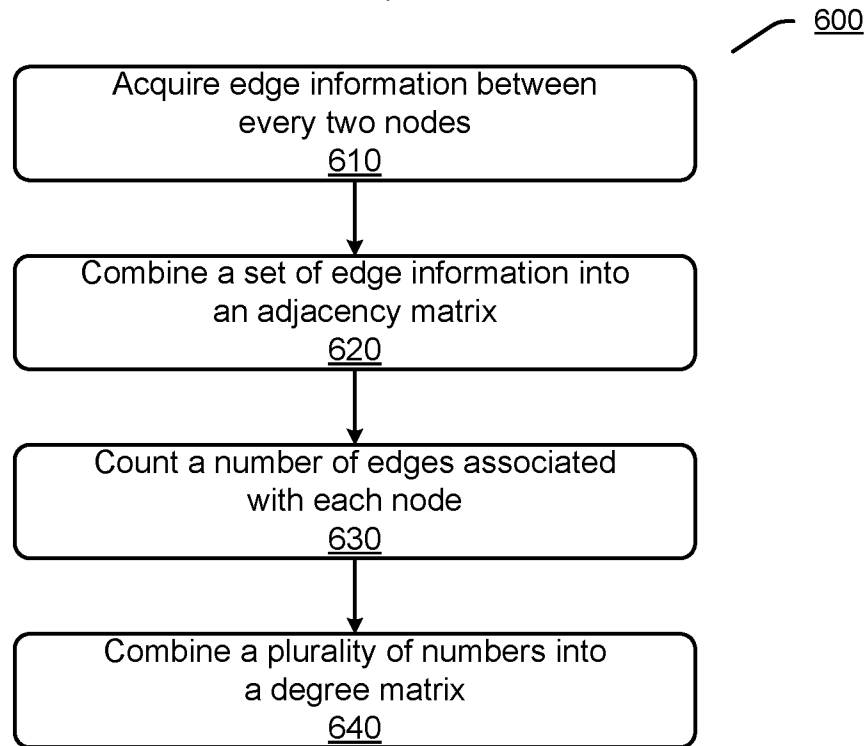
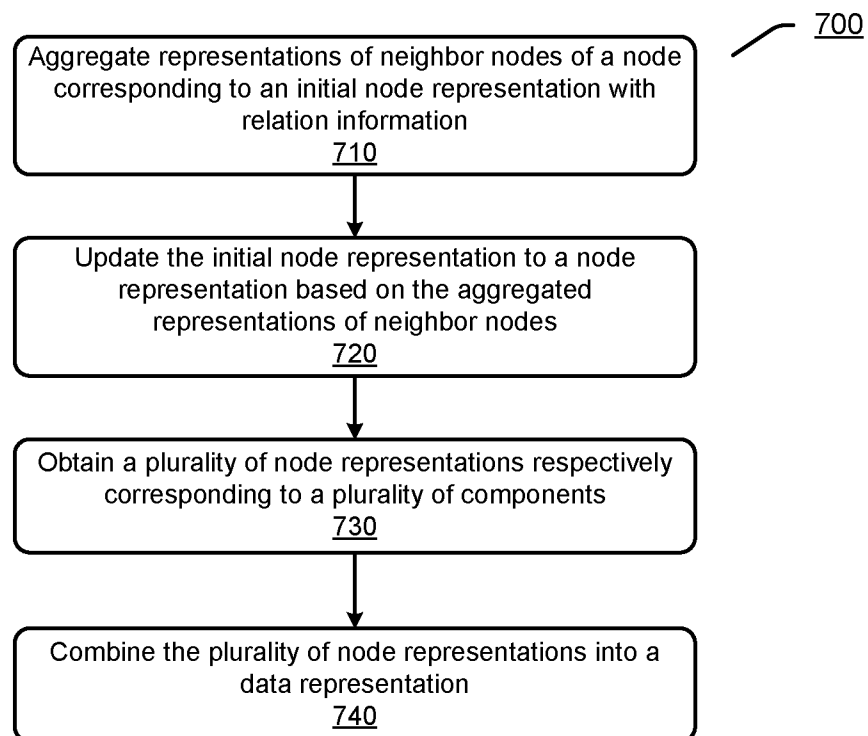


FIG 5

6/9

**FIG 6****FIG 7**

7/9

800

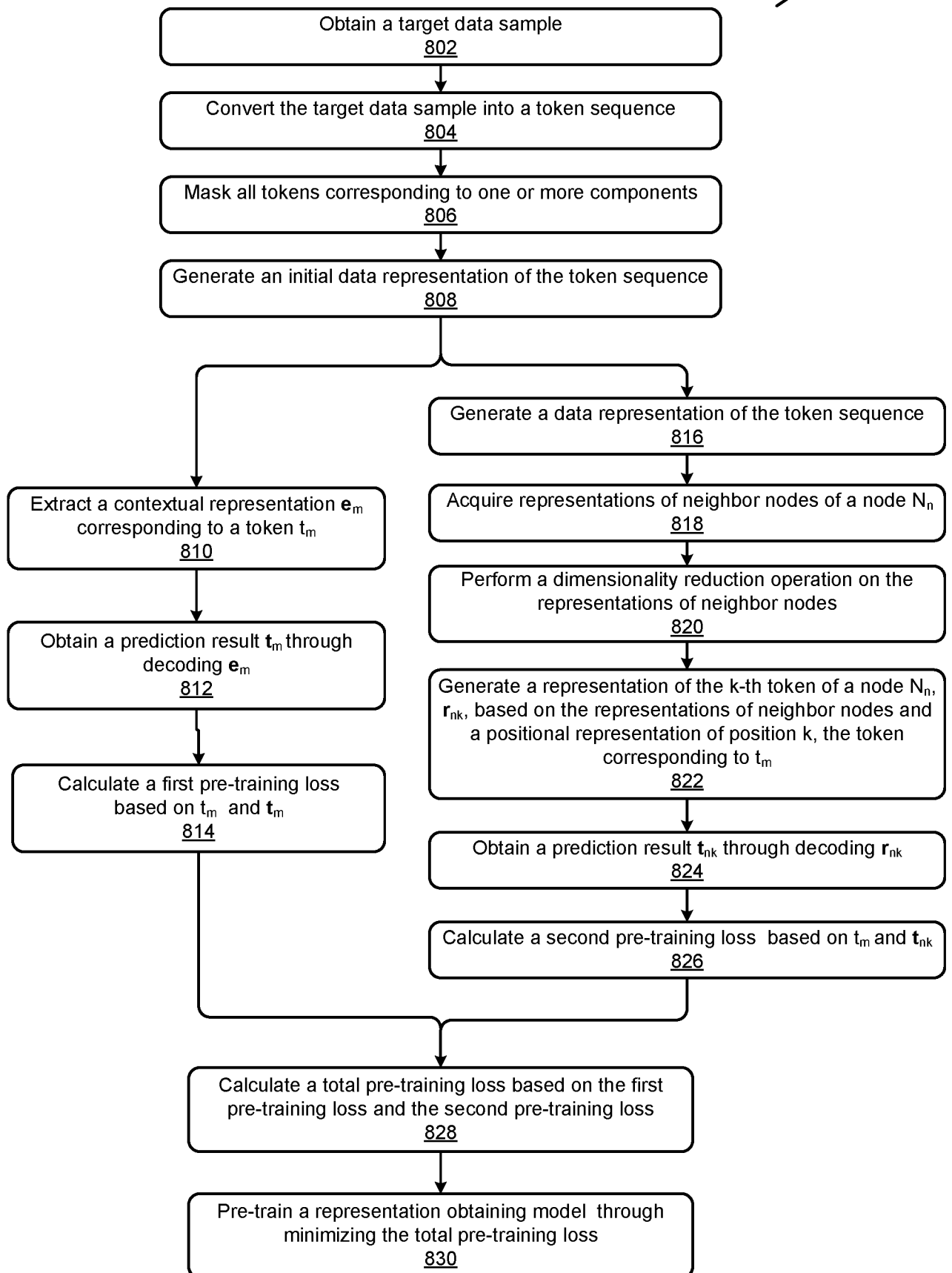


FIG 8

8/9

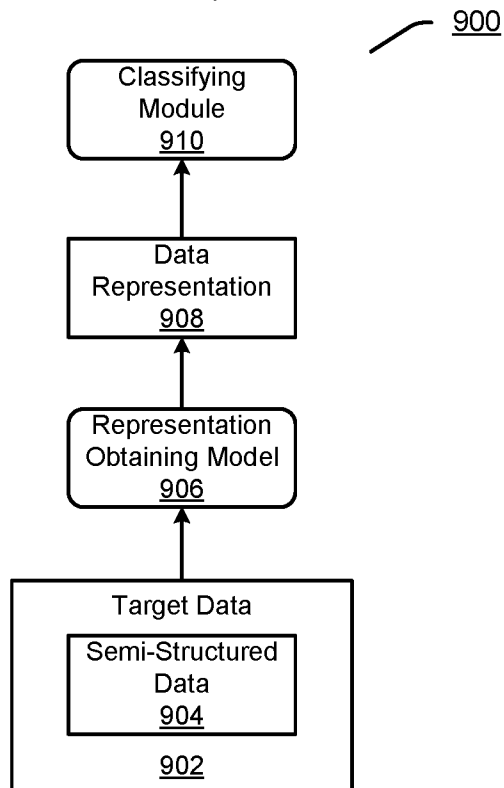


FIG 9

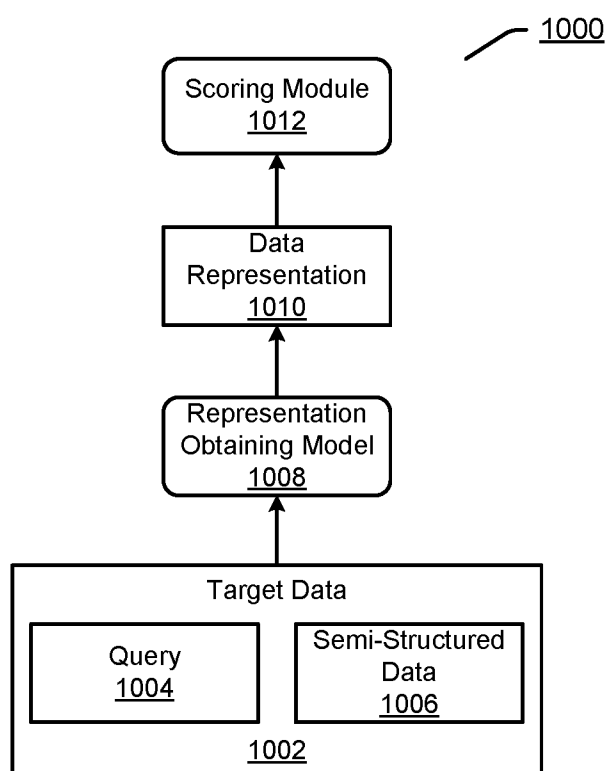
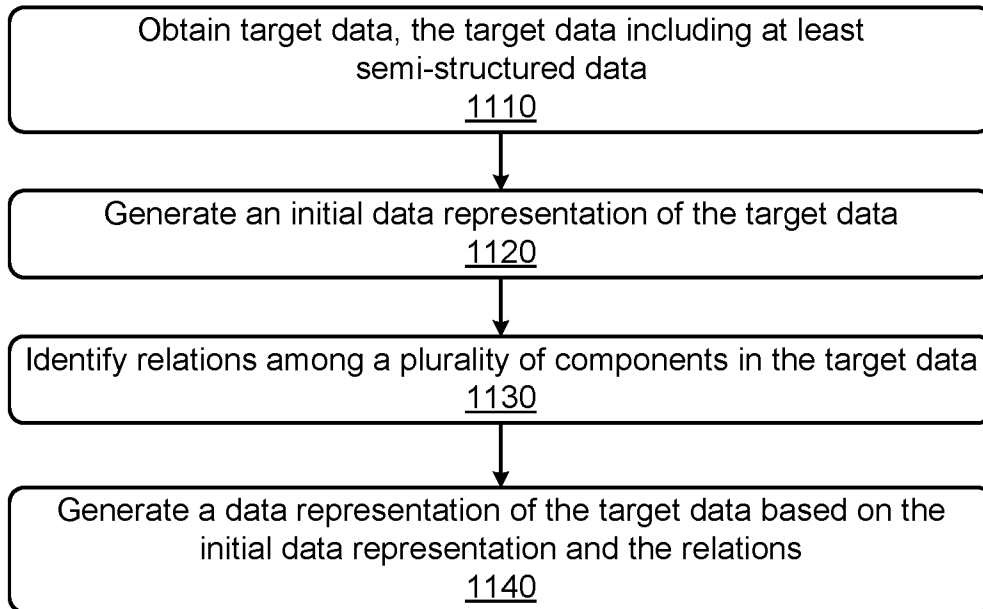
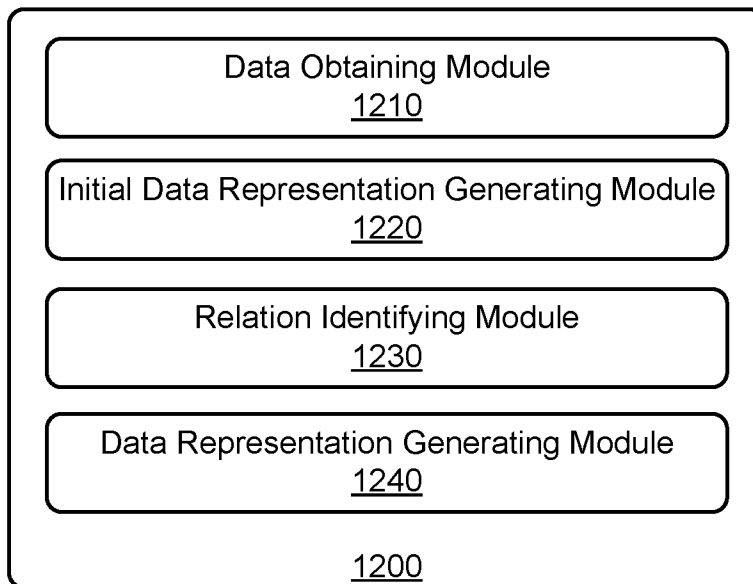
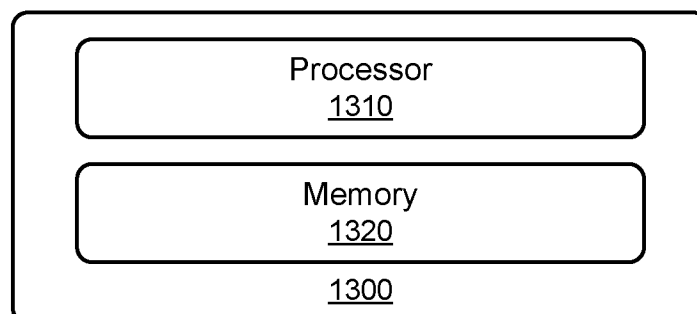


FIG 10

9/9

1100

**FIG 11****FIG 12****FIG 13**

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2021/034689

A. CLASSIFICATION OF SUBJECT MATTER

INV. G06F16/84 G06F16/332
ADD.

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 G06N

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)

EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	PETER BUNEMAN ED - ASSOCIATION FOR COMPUTING MACHINERY: "Semistructured data", PROCEEDINGS OF THE SIXTEENTH ACM SIGACT-SIGMOD-SIGART SYMPOSIUM ON PRINCIPLES OF DATABASE SYSTEMS. PODS 1997. TUCSON, MAY 12 - 14, 1997; [PROCEEDINGS OF THE ACM SIGACT-SIGMOD-SIGART SYMPOSIUM ON PRINCIPLES OF DATABASE SYSTEMS. PODS 1997], NEW YORK, A, 1 May 1997 (1997-05-01), pages 117-121, XP058284629, DOI: 10.1145/263661.263675 ISBN: 978-0-89791-910-4 abstract page 118 -----	1-15



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents :

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

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

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of 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

23 August 2021

Date of mailing of the international search report

07/09/2021

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Rameseder, Jonathan