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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
13 March 2025 (13.03.2025)

WIPO | PCT

(10) International Publication Number
WO 2025/053765 A1

(51) International Patent Classification:
G06F 16/20 (2019.01) *G06N 3/08* (2023.01)
G06F 16/30 (2019.01) *G06N 20/00* (2019.01)
G06F 40/284 (2020.01) *G06F 16/00* (2019.01)

(21) International Application Number:
PCT/NZ2024/050104

(22) International Filing Date:
05 September 2024 (05.09.2024)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
2023902868 06 September 2023 (06.09.2023) AU

(71) Applicant: **XERO LIMITED** [NZ/NZ]; 19-23 Taranaki Street, Te Aro, Wellington, 6011 (NZ).

(72) Inventors: **WINTER, Leah**; C/o - Xero Limited, 19-23 Taranaki Street, Te Aro, Wellington, 6011 (NZ). **QIAN, Tracy**; C/o - Xero Limited, 19-23 Taranaki Street, Te Aro, Wellington, 6011 (NZ). **DESAI, Khantil**; C/o - Xero Limited, 19-23 Taranaki Street, Te Aro, Wellington, 6011 (NZ). **ANWAR, Muhammed**; C/o - Xero Limited, 19-23 Taranaki Street, Te Aro, Wellington, 6011 (NZ). **CHEAH, Soon-Ee**; C/o - Xero Limited, 19-23 Taranaki Street, Te Aro, Wellington, 6011 (NZ). **XIE, Holly**; C/o - Xero Limited, 19-23 Taranaki Street, Te Aro, Wellington, 6011 (NZ). **SOBRAL, Thiago**; C/o - Xero Limited, 19-23 Taranaki Street, Te Aro, Wellington, 6011 (NZ).

(74) Agent: **FB RICE PTY LTD**; Level 11, Waterfront Place, 1 Eagle Street, Brisbane, Queensland 4000 (AU).

(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, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO,

(54) Title: SYSTEMS, METHODS AND COMPUTER-READABLE MEDIA FOR DETERMINING ATTRIBUTES OF DOCUMENTS

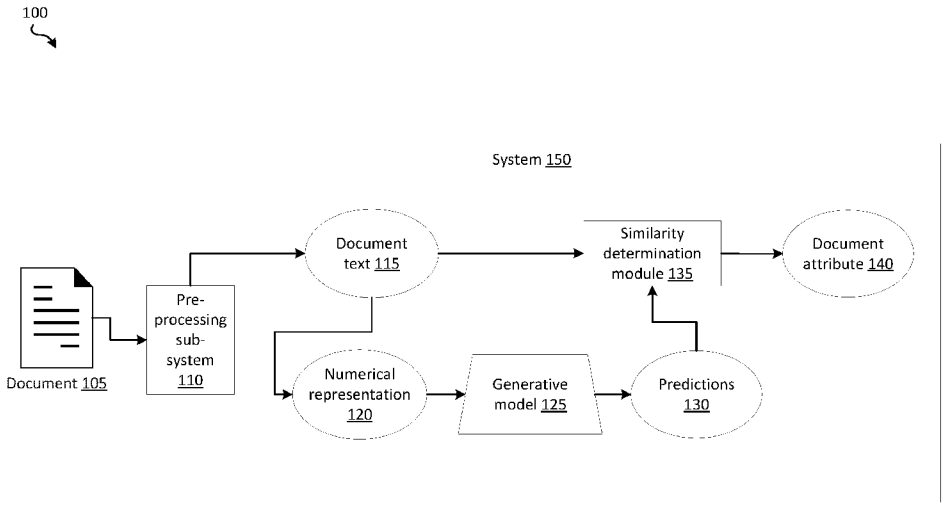

Fig. 1

(57) Abstract: Described embodiments involve determining a numerical representation of the document for which a document attribute is sought to be determined, providing the numerical representation to a generative model to determine a generative model prediction of the document attribute, determining a candidate document character string of the document text that is sufficiently similar to the generative model prediction, determining the document attribute based on the generative model prediction and/or the candidate document character string; and outputting the determined document attribute.

[Continued on next page]

WO 2025/053765 A1

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, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, 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, ME, 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).

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*

Systems, methods and computer-readable media for determining attributes of documents

Cross Reference to Related Application

[0001A] The present application claims priority from Australian Provisional Patent Application No. 2023902868 filed on 6 September 2023, the contents of which are incorporated herein by reference in their entirety.

Technical Field

[0001B] Embodiments generally relate to systems, methods and computer-readable media for determining or identifying attributes, such as entity identifiers, associated with documents.

Background

[0002] Manually reviewing documents to extract information from them can be a time-intensive, arduous and error-prone process. For example, documents need to be visually inspected to extract specific information from the documents. After the visual inspection, the determined information needs to be manually entered into a computer system. Data entry processes are often prone to human error. If a relatively large number of documents need to be processed, significant time and resources may be expended to ensure that complete and accurate data entry has been performed.

[0003] It is desired to address or ameliorate some of the disadvantages associated with prior methods and systems for extracting information from documents, or at least to provide a useful alternative thereto.

[0004] Any discussion of documents, acts, materials, devices, articles or the like which has been included in the present specification is not to be taken as an admission that any or all of these matters form part of the prior art base or were common general

knowledge in the field relevant to the present disclosure as it existed before the priority date of each of the appended claims.

Summary

[0005] Described embodiments relate to a method comprising: determining document text comprising a plurality of document words, the document text being associated with a document for which a document attribute is sought to be determined; determining a numerical representation of the document text; providing the numerical representation to a generative model to determine a generative model prediction of the document attribute, the generative model prediction comprising a predicted character string, the predicted character string comprising one or more predicted words; determining a candidate document character string of the document text that is sufficiently similar to the predicted character string, the candidate document character string comprising one or more of the plurality of document words; determining the document attribute based on the predicted character string and/or the candidate document character string; and outputting the determined document attribute.

[0006] The determined document attribute comprises: (i) the predicted character string; (ii) the candidate document character string; or (iii) one or more predicted words of the predicted character string and one or more document words of the candidate document character string.

[0007] Determining the candidate document character string may comprise: for each predicted word and document word pair, determining a similarity measure indicative of the similarity of the predicted word and document word of the pair; for each predicted word, determining the predicted word and document word pair with a highest similarity measure; and selecting the document word of the predicted word and document word pair as a document word of the candidate document character string.

[0008] For example, determining the similarity measure for a predicted word and document word pair may comprise determining whether the predicted word and

document word pair meets a first, second or third similarity threshold, wherein the first threshold is that the predicted word corresponds exactly with the document word, the second threshold is that the predicted word is of a threshold length and is within a threshold edit distance of the document word, and the third threshold is that the predicted word is of a threshold length and is a substring of the document word; and responsive to determining that the predicted word and document word pair meets the first, second or third threshold, determining the similarity measure as a first similarity measure, a second similarity measure, or a third similarity measure, respectively.

[0009] The first similarity measure may be indicative of a greater level of similarity than the second similarity measure and the third similarity measure, and the second similarity measure is indicative of a greater level of similarity than the third similarity measure.

[0010] In some embodiments, determining the similarity measure for a predicted word and document word pair comprises: determining whether the predicted word corresponds to the document word of the predicted word and document word pair; and responsive to determining that the predicted word corresponds to the document word, determining the predicted word and document word pair as the predicted word and document word pair with the highest similarity measure for the predicted word. For example, determining that the predicted word corresponds to the document word may comprise determining that the predicted word and document word pair meet a first similarity threshold, and assigning the predicted word and document word pair a first similarity measure.

[0011] In some embodiments, determining the similarity measure for a predicted word and document word pair comprises: determining whether the predicted word and document word pair meets a second similarity threshold, wherein the second similarity threshold is that the predicted word is of a threshold length and is within a threshold edit distance of the document word; and responsive to the predicted word and document word pair meeting the second similarity threshold, assigning the predicted word and document word pair a second similarity measure.

[0012] In some embodiments, determining the similarity measure for a predicted word and document word pair comprises: determining whether the predicted word and document word pair meets a third similarity threshold, wherein the third similarity threshold is that the predicted word is of a threshold length and is a substring of the document word; and responsive to the predicted word and document word pair meeting the third similarity threshold, assigning the predicted word and document word pair a third similarity measure.

[0013] Responsive to determining that each of the predicted words of the predicted character string corresponds to a respective document word of the candidate character document string, the predicted character string may be determined as the document attribute. Responsive to the highest similarity measure for predicted word and document word pair being the first similarity measure, the document attribute may be determined to comprise the predicted word of the predicted word and document word pair. Responsive to the highest similarity measure for predicted word and document word pair being the second similarity measure, the document attribute may be determined to comprise the predicted word of predicted word and document word pair. Responsive to the highest similarity measure for predicted word and document word pair being the third similarity measure, the document attribute may be determined to comprise the document word of predicted word and document word pair.

[0014] In some embodiments, determining document text associated with a document for which a document attribute is sought to be determined comprises: removing any non-alphanumeric characters and/or converting all text to lower case.

[0015] In some embodiments, determining text comprises tokenizing the document text into one or more document tokens, and wherein determining a numerical representation of the document text comprises determining a numerical representation of the one or more document tokens.

[0016] In some embodiments, the method further comprises providing the determined document attribute to a reconciliation module; and reconciling a transaction associated with the document using the determined document attribute.

[0017] In some embodiments, the method further comprises providing the determined document attribute to an entity attribute management module; and (i) creating a new entity profile in a database using the determined document attribute, or (ii) updating an existing entity profile in the database using the determined document attribute.

[0018] Some embodiments relate to a system comprising: one or more processors; and memory comprising computer executable instructions, which when executed by the one or more processors, cause the system to perform any one or more of the described methods.

[0019] Some embodiments relate to a computer-readable storage medium storing instructions that, when executed by a computer, cause the computer to perform any one or more of the described methods.

[0020] Throughout this specification the word "comprise", or variations such as "comprises" or "comprising", will be understood to imply the inclusion of a stated element, integer or step, or group of elements, integers or steps, but not the exclusion of any other element, integer or step, or group of elements, integers or steps.

Brief Description of Drawings

[0021] Various ones of the appended drawings merely illustrate example embodiments of the present disclosure and cannot be considered as limiting its scope.

[0022] Figure 1 is a schematic diagram of a system for determining document attributes, according to some embodiments;

[0023] Figure 2 is a schematic of an example of the determination of a document attribute based on a comparison of a generative model prediction and a candidate document character string;

[0024] Figure 3 is a block diagram of the system of Figure 1 for determining document attributes, according to some example embodiments; and

[0025] Figure 4 is a process flow diagram for determining document attributes, according to some embodiments.

Detailed description

[0026] Embodiments generally relate to systems, methods and computer-readable media for determining or identifying attributes, such as entity identifiers, associated with documents.

[0027] Documents in the real world may be created using an automated process or a combination of automated processes and manual processes. For example, an invoice by a business may be created using a predefined template and manual population of the predefined template. Documents may include a printed matter, such as images, logos, printed text or formatting artefacts such as tables, lines or patterns. Documents may also include handwritten annotations. Businesses may have their distinct templates for creating documents. One business may have more than one distinct template for creating documents. For example, a business may have a first template used for creating invoices and a second template for creating quotes. Thus, there exists variability in the structure and overall appearance of documents created by businesses or organisations. Accordingly, automated text extraction may be inaccurate at extracting specific document attributes, such as a document issuer, or other entity name, for example.

[0028] Generative models may be capable of generating, based on inputs such as a documents, document text, or a numerical representation of document text. However,

generative models, by their nature, can predict sequences that don't necessarily exist in their prompt or input, such as the document text. This can result in messy or inaccurate attribute predictions, such as a misspelling of an entity name, for example "Woolwurths" instead of "Woolworths", improper formatting, such as "e-bay" instead of "eBay", or entirely inappropriate predictions that seem out of context, such as "Wheelies". In some cases, a generative model, could potentially generate outputs that could expose entities or individuals to excessive amounts of risk, such as data security risks.

[0029] The present disclosure involves using a generative model to determine or extract attributes, such as entity identifier, from documents. For example, the described embodiments may be used to determine business or organisation identifiers, such as names, from financial documents, such as financial statements, or business records, invoices etc. The extraction of such attributes from financial documents can be used by various modules of an accounting system. For example, a reconciliation module may use the determined attributes to automatically reconcile financial transactions, or may present them in a user interface to a user as a suggestion for aiding in reconciling a particular transaction. Similarly, determination of such document attributes, such as entity names or identifiers, can be used to automatically create or update, or suggest to a user via a user interface to create or update an entry in a register or database, such as a contact list or address book.

[0030] To mitigate the chance of the generative model predicting attributes that are inappropriate, inaccurate and/or out of context, the predicted attribute of a document, as determined by the generative model, is compared with the text of the document to ensure that the predicted attribute, or at least a similar form of the predicted attribute, appears in the text of the document. Only predicted attribute(s) that are sufficiently similar to text or a character string from the document will be accepted as the determined attribute of the document, and output to other or downstream processes, such as a reconciliation module. In other words, the determined attribute of the document will be a character string that exactly matches, or is sufficiently similar to, text/a character string of the document.

[0031] An exact match of the predicted attribute with text/a character string from the documents provides a fail-safe, ensuring that the output/ determined attribute of the document actually appears in the document.

[0032] However, by allowing for or accepting predicted attributes that are sufficiently similar to text/character strings from the documents accommodates for the imprecise nature of generative models, which may be prone to make predictions that include errors, such as generative hallucinations, misspellings, transcription errors (e.g., introduced during an OCR process), for example. This may result in an increased number of predictions being accepted and output for other processes, and/or may increase accuracy in that transcription errors or typos can be accommodated.

[0033] **Figure 1** is a schematic diagram of a process 100 for determining document attributes, according to some embodiments. Figure 1 depicts a system 150 configured to receive and process documents to automatically determine document attributes of the documents, which may be used for further processes and/or stored in databases. For example, the system 150 may ingest financial documents, such as bank statements, invoices, receipts etc., and pass relevant document attributes to an account or financial application for financial management, such as transaction reconciliation, payment tracking etc. The system 150 may be an accounting system, or an attribute determination system, for example.

[0034] According to Figure 1, a document 105 is provided to a pre-processing subsystem 110 of the system 150. The document 105 comprises text, some of which is indicative of one or more attributes of the document 105. For example, the document 105 may be an invoice and the text may comprise amongst others, a character string that is indicative of an issuer of the invoice. The pre-processing subsystem 110 determines document tokens 115. The system 150 determines a numerical representation 120 of the document text or document text tokens. Each of the document text tokens may be indicative of one full or partial word of the document 105. For example, the text tokens may be considered word pieces. The numerical representation 120 may be an embedding representation of the tokens or of the plain text of the

document, for example. In some embodiments, the pre-processing subsystem 110 may perform one or more preparation processes, such as text recognition, text filtering and/or text string preparation.

[0035] Based on the numerical representation 120, a generative model 125 generates a generative model prediction or predicted character string of the document attribute, which is comprised of one or more prediction words 130.

[0036] The prediction word(s) 130 of the predicted character string are then compared with candidate document words by the similarity determination module 135 to determine their similarity with one another and whether it falls within a predetermined threshold. Specifically, words of the predicted character string are compared with document word. In some embodiments, to meet or comply with the predetermined threshold, each predicted word must: (i) correspond exactly with a document word; (ii) be of a threshold length (e.g., 3 characters long), and be within a threshold edit distance of a document word; or (iii) be of a threshold length (e.g., 3 characters long), and be a substring of a document word. Based on this comparison, the similarity determination module 135 determines whether a suitable or appropriate document attribute can be determined and output.

[0037] A similarity measure of a predicted word and document word pair is based on whether or not the words of the string pair pass the first, second or third thresholds. Predicted character strings with word(s) which do not pass any of the first, second or third thresholds are considered as being not sufficiently similar and are not used to determine the document attribute. A greater weighting is given to words which pass the first threshold, with a lesser weighting being given to words which pass the second threshold, and an even lesser weighting being given to words which pass the third threshold. Accordingly, similarity measures of predicted and document word pairs can be rated or ranked against one another to determine a most suitable or best match.

[0038] The outputted or determined document attribute may comprise a character string comprising (i) all of the prediction words (for example, where the prediction

words corresponded exactly with document words, or were substrings of document words), (ii) all of the document words (where less than a threshold number of edits were required to convert the prediction words into the document words); or (iii) one or more prediction words and one or more document words (where one or more predictions words were exactly the same as or substrings of the document words, and one or more predictions words required less than a threshold number of edits to convert them into document words).

[0039] Referring now to Figure 2, there is illustrated an example of how the similarity determination module 135 may be configured to determine the document attribute 140 to be outputted based on a comparison of predicted words of a generative model prediction and document words of the document text.

[0040] As shown at stage 1, in this example, the system 150 has determined the generative model prediction to be the predicted character string: "AMPN Supplis Products". This generative model prediction is to be compared with the document character string: "AMPM Supplies products@ampm.com".

[0041] The system 150 pre-processes both the predicted character string and the document character string to convert all characters to lowercase and to remove all non-alphanumeric characters, such as "@" and ".". The pre-processed characters strings are then: "ampn supplis products" and "ampm supplies productsampmcom", as shown in stage 1. At this stage, a vote for each word ("Token" in Fig. 2) is initialised as zero.

[0042] To determine whether a document attribute can be determined from the predicted character string and the candidate document character string, the pre-processed predicted words of the predicted character string are then compared with each of the pre-processed candidate document words to determine their similarity with one another and whether it falls within a predetermined similarity threshold.

[0043] If the predicted word corresponds with or is exactly the same as a candidate document word, the predicted word is retained as a word of the determined document

attribute. Thus, a first similarity threshold would be that the predicted word is the same as the candidate document word. This is not the case with the example of Figure 2.

[0044] In this example, and as shown at stage 2, for each predicted word, the similarity determination module 135 determines whether the respective predicted word meets a threshold character length (e.g., 3 characters), and is within a threshold edit distance of a document word of the candidate document string (e.g., a second similarity threshold). This edit distance assessment stage is performed for any predicted words which do not meet the first similarity threshold.

[0045] The edit distance measure may be the Damerau Levenstein edit distance calculation. This involves determining a number of edits (in the form of deletions, insertions, substitutions or transpositions) required to convert or transform each predicted word of the predicted character string into a document word of the candidate document character string. In this way, the similarity determination module 135 is effectively applying a "fuzzy logic" matching process.

[0046] As shown in stage 2, the system 150 determines that: 1 edit is required to transform "ampn into ampm", 7 edits are required to transform "ampn" into "supplies" and 11 edits are edits are required to transform "ampn" into "productsaampmcom"; 6 edits are required to transform "supplis" into "ampm", 1 edit is required to transform "supplis" into "supplies", and 13 edits are required to transform "supplis" into"productsaampmcom"; and 8 edits are required to transform "products" into ampm", 7 edits are required to transform "products" into "supplies" and 7 edits are required to transform "products" into "productsaampmcom".

[0047] Accordingly, in this example where the first threshold has not been met, a second similarity measure (e.g., edit distance measure) is determined for each word pair of the of the predicted character string and the candidate document string, the second similarity measure being indicative of a number of edits required to convert the predicted word into the candidate document word. For each word pair, a lowest edit distance measure is selected, representing the best match for the second similarity

measure. The lowest edit distance of the pair is compared with the second similarity threshold (e.g., an edit distance threshold, such as less than or equal to 1), and if it meets the threshold, the pair is selected, meaning the predicted word is to be substituted for the document word of the pair.

[0048] In this example, the second similarity threshold is "1". So, for the predicted word "AMPN", the document word "AMPM" is selected as a suitable substitution word as the pair have the lowest edit distance measure of the document words with the predicted word "AMPN", and the lowest edit distance measure meets the second similarity threshold of "1". Similarly, for the predicted word "Supplis", the document word "Supplies" is selected as a suitable substitution word as the pair have the lowest edit distance measure of the document words with the predicted word "Supplis", and the lowest edit distance measure meets the second similarity threshold of "1". Conversely, for the predicted word "Products", both document words "Supplies" and products@ampn.com share the lowest edit distance of "7", but this does not meet the second similarity threshold and accordingly, neither are selected as suitable substitutions for the predicted word "Products".

[0049] As shown at stage 3 of the example, the system 150 performs a substring matching process. In this example, for each predicted word that has not pass the first or second similarity threshold, the similarity determination module 135 determines whether the respective predicted word meets a threshold character length (e.g., 3 characters), and is a substring of a document word of the candidate document string (e.g. a third similarity threshold). This substring matching stage is performed for any predicted words which do not meet the first and second similarity thresholds. As shown at stage 3, the word "products" is a substring of "productsampmcom" and is longer than 3 characters. Accordingly, the predicted token "Products" is selected as an appropriate word for the document attribute.

[0050] The system 150 therefore determines or outputs "AMPM Supplies Products" as the determined document attribute.

[0051] **Figure 3** is a block diagram of a computer network 300 comprising the system 150 for determining document attributes, according to some embodiments. The system 150 such as an accounting system (or a system providing an accounting platform is in communication with one or more client devices 310, database(s) 340, and/or financial institute systems 345 across a communications network 330.

[0052] Client device 310 may be a computing device such as a personal computer, laptop computer, desktop computer, tablet, smart phone, or any other suitable and/or similar type of computing device, for example. The client device 310 comprises one or more processor(s) 212, memory 216, communications circuitry 214, and device I/O 220, for example.

[0053] The processor(s)312 may comprise one or more microprocessor circuitry, central processing units (CPUs), application specific instruction set processor (ASIPs) circuitry, application specific integrated circuits (ASICs) or other processor circuitry capable of reading and executing instruction code.

[0054] The communications interface 314 may comprise a combination of network interface hardware and network interface software suitable for establishing, maintaining and facilitating communications over a relevant communication channel. The communications interface 314 may work in conjunction with one or more communications program code (not shown), to facilitate communications over the relevant communication channel.

[0055] Memory 316 comprises one or more volatile or non-volatile memory types. For example, memory circuitry 316 may comprise one or more of random-access memory (RAM), read-only memory (ROM), electrically erasable programmable read-only memory (EEPROM) or flash memory. The memory 316 is configured to store program code accessible by the processor(s) 312. The program code comprises executable program code modules. In other words, the memory 316 is configured to store executable code modules configured to be executable by the processor(s) 312. The executable code modules, when executed by the processor(s) 312 cause the client

device 310 to perform certain functionality to cooperate with the system 150, as described in more detail below. For example, the memory 316 may comprise application software 318.

[0056] Application software 218 may be one or more of a web browsing application (not shown), attribute determination application (not shown) or any other type of application that may enable a user of the client device 310 to avail of the functionalities of the system 150.

[0057] Device I/O 320 may comprise a display monitor, such as an LED screen, LCD screen, touch screen or any other suitable display. Device I/O 320 may comprise one or more input devices, such as a keyboard, mouse, touch screen, and/or camera.

[0058] The network 330 may include, for example, at least a portion of one or more networks having one or more nodes comprising communication circuitry configured to transmit, receive, forward, generate, buffer, store, route, switch, process, or a combination thereof, etc. one or more messages, packets, signals, some combination thereof, or so forth. The network 330 may include, for example, one or more of: a wireless network, a wired network, an internet, an intranet, a public network, a packet-switched network, a circuit-switched network, an ad hoc network, an infrastructure network, a public-switched telephone network (PSTN), a cable network, a cellular network, a satellite network, a fibre-optic network, some combination thereof, or so forth.

[0059] The database 340, which may form part of or be local to the system 150, or may be remote from and accessible to the system 150, for example, via the communications network 330. The database 340 may be configured to store data associated with the system 150. The database 340 may be a centralised database. The database 340 may be a mutable data structure. The database 340 may be a shared data structure. The database 340 may be a data structure supported by database systems such as one or more of PostgreSQL, MongoDB, and/or ElasticSearch. The database 340 may be configured to store a current state of information or current values associated

with various attributes (e.g., "current knowledge"). For example, database 340 may store one or more document files, text datasets, token datasets, machine learning models, machine learning training data sets, document attributes and/or document attribute predictions.

[0060] The system 150 may also be arranged to communicate with financial institute systems 345 or other third-party financial systems (not shown) to communicate document attributes or other document data, and/or receive documents or document text for determining document attributes. For example, in some embodiments, the system 150 may be arranged to receive documents or document text associated with users or user entities associated with the financial institute systems 345. The financial institute systems 345 may provide an Application Programming Interface (API) to securely transfer documents or document text. The APIs may be secured using authentication and encryption mechanisms and the documents or document text may be referred to as invoices, quotes, bills, credit notes, remittance or invoice data, quote data, bill data, credit note data or remittance data. Invoice data, quote data, bill data, credit note data or remittance data may comprise information relating to an issuer of a particular document associated with the data.

[0061] The system 150 may be one or more computing devices, such as one or more servers. The system 150 may comprise one or more processor(s) 352, communications interface 354, and memory 356. As mentioned above, the system 150 may be in communication with the client device(s) 310, database 340 and/or financial institute systems 345 and/or other third-party systems (not shown), over network 330, for example.

[0062] The processor(s) 352 may comprise one or more microprocessor circuitry, central processing units (CPUs), application specific instruction set processor (ASIPs) circuitry, application specific integrated circuits (ASICs) or other processor circuitry capable of reading and executing instruction code.

[0063] The communications interface 354 may comprise a combination of network interface hardware and network interface software suitable for establishing, maintaining and facilitating communications over a relevant communication channel. The communications interface 314 may work in conjunction with one or more communications program code (not shown), to facilitate communications over the relevant communication channel.

[0064] Memory 356 may comprise one or more volatile or non-volatile memory types. For example, memory 356 may comprise one or more of random-access memory (RAM), read-only memory (ROM), electrically erasable programmable read-only memory (EEPROM) or flash memory. The memory 356 is configured to store program code accessible by the processor(s) 352. The program code comprises executable program code modules. In other words, the memory 356 is configured to store executable code modules configured to be executable by the processor(s)352. The executable code modules, when executed by the processor(s) 352 cause the system 150 to perform certain functionality, as described in more detail below. For example, the memory 316 may comprise a data handling module 358, a training module 360, the generative model 125, a pre-processing module 364, a tokenization module 366, a numerical representation model 368, a similarity determination module 135, an attribute output module 372, a transaction reconciliation module 374 and/or an entity record management module 376.

[0065] The data handling module 358 may be configured to receive, request and handle data to be used by the system 150 in determining document attributes. Data handling module 358 may be configured to communicate outputs from the attribute determination system 150, for example to client device(s) 310 and/or database 340. Data handling module 358 may work in conjunction with the communications interface 354 to facilitate the receipt, request, and/or transfer of data.

[0066] The training module 360 may be configured to train one or more machine learning models, such as the generative model 125. The training module 260 may be configured to run a training program to train one or more machine learning models to

determine document attributes based on numerical representations of document text or tokens indicative of document text. The models being trained by the training module 260 may be pre-trained. For example, the generative model 125 may be initialised by the training module 260 from a pre-trained encoder-decoder model, such as a pre-trained base-T5 model available within the Hugging face library (https://huggingface.co/).

[0067] To train the generative model 125, the training module 360 may determine a training dataset comprised of training documents from a corpus of documents. The training documents each comprise document text that includes one or more words or character strings indicative of a document attribute to be determined by the generative model 125. For example, each of the training documents may include within the body of their text a word or character string indicative of an entity that generated or provided the document, such as an issuer of an invoice. Each training document has an associated label indicating the document attribute. The training dataset used for training the generative model 125 may be stored in database 340 or accessed from financial institute systems 345. In some embodiments, the training dataset may be a subset of the documents stored in database 340 or financial institute systems 345.

[0068] During the training process, the generative model 125 may be tested on its ability to correctly predict a document attribute of the training dataset. During the training process, the training module 360 may be configured to update weightings of the model's nodes based on the model's predictions. The training process may be performed over several epochs.

[0069] The generative model 125 is configured to receive as an input, document text, document text tokens, a numerical representation of the document text or a numerical representation of the document text tokens and generate or otherwise provide a prediction of a document attribute. For example, the document attribute may be a generator or issuer of the document from which the document text has been determined. The generative model 125 may be determined by the training module 360.

[0070] According to some embodiments, the generative model 125 may be a machine learning model. The machine learning model may be an artificial neural network, such as a recurrent neural network.The artificial neural network may comprise a transformer architecture. In some embodiments, the generative model 125 may be a pre-trained base-T5 model available within the Huggingface library that has been further tuned using the downstream task as performed by the training model 360.

[0071] In some embodiments, the generative model 125 may not be provided with all of the document text comprised within a document, or all document tokens determined from the document text. In some embodiments, the generative model 125 may be provided the first n number of tokens of the document, such as the first 512 tokens.

[0072] The pre-processing module 364 may be configured to prepare a document 105, or document text for use by the system 150 for determining document attributes. In some embodiments, the pre-processing module 364 may be configured to process the document text to determine pre-processed document text for tokenisation. The pre-processed document text may have had all non-alphanumeric symbols removed. In some embodiments, the pre-processed document text may have all text converted to lower-case. In some embodiments, the document text and the pre-processed document text may be stored, for example in database 360. The pre-processing module 364 may form part of the pre-processing subsystem 110.

[0073] The tokenization module 366 may be configured to determine one or more tokens from a provided document, or in particular the text information of the provided document. The text or character strings of the records may be converted into tokens or tokenized using a tokenizer such as WordPiece ("Japanese and Korean Voice Search (Schuster et al. 2012)"). In some embodiments, the document or the document text information may comprise one or more n-grams. N-grams are a series of n adjacent elements in a document or set of text information that may include words, numbers, symbols, and punctuation. In some embodiments, certain n-grams may be collapsed into a single token to better capture the context or meaning of the n-gram. For example, the bigrams "credit note" and "credit card" have specific meanings that may

be contextually relevant given the nature of the document or document text information, and accordingly, these bigrams may each be collapsed into a single representative token. The tokenization module 366 may form part of the pre-processing subsystem 110.

[0074] The numerical representation model 368 may be configured to receive tokens, such as generated or otherwise determined by tokenization module 366, and provide as an output, a numerical representation of the tokens. For example, the numerical representation model 368 may generate a numerical representation in the form of a vector using a vectorisation technique suitable for natural language processing tasks.

[0075] In some embodiments, the numerical representation model 368 may be configured to generate numerical representations of tokens using a corpus of words or data strings as may be stored in the database 340 and/or stored in the system 150, and/or accessible to the attribute determination module 363. For example, the documents and/or records used to generate the corpus (e.g., a learned vocabulary) may comprise invoices/bills, credit notes, receipts/spend money transactions etc. In some embodiments, the corpus may be generated from example documents of a dataset to be used to train the generative model 125. The corpus may be a corpus of tokens such as lemmas. Each token of the corpus is associated with a respective vector or word embedding. For example, each vector may have a dimension of 512. The numerical representation model 368 may be configured to determine a vector for each token using the corpus, and sum or average the vectors to determine the numerical representation of the tokens, or in some embodiments, a numerical representation of the provided document. For example, the numerical representation model 368 may comprise a WordPiece encoder ("Japanese and Korean Voice Search (Schuster et al. 2012)").

[0076] In embodiments where a token to be converted into a numerical representation does not appear in the corpus, (for example, has not been seen before), numerical representation model 368 may be configured to assign the token a vector used for unknown tokens and use that vector in determining or generating the numerical representation. For example, the numerical representation model 368 may be

configured to swap the token for an "UNKNOWN" token which has a corresponding vector in the corpus.

[0077] In some embodiments, a threshold number of tokens is provided to the numerical representation model 368. Any suitable number may be selected for the threshold. For example, the threshold may be 400, 500, or 512 tokens. Where a document comprises more than the threshold number of tokens, any tokens in excess of the threshold are discarded or not accepted as an input to the numerical representation model 368. In cases where a document comprises less than the threshold number of tokens, a vector of the document may be padded, for example, with zero. Determination of a suitable number of tokens may involve balancing accuracy/effectiveness, processing speeds and/or a number and/extent to which documents need to be padded. For example, if the threshold is too high, a relatively large number of documents may need to be padded, which may impact speed for little or no improved accuracy. On the other hand, if the threshold is too low, too many tokens may be discarded and relevant information may not be taken into account when the numerical representation is generated.

[0078] In some embodiments, the numerical representation model 368 may comprise a trained neural network to generate the word embeddings or vectors corresponding to each token. In some embodiments, the numerical representation model 268 may include one or more language models such as the Bidirectional Encoder Representations from Transformers (BERT). The numerical representation model 368 may form part of the pre-processing subsystem 110. The similarity determination module 135 is configured to determine whether the words of predicted character strings and document words are sufficiently similar. Where a determination is made that the predicted words of a predicted character string are sufficiently similar to candidate document words of the document text, the system 150 determines the document attribute for the document based on the predicted character string and/or the candidate document words. The determined document attribute comprises or is the predicted character string, the candidate document words; or one or more predicted words of the predicted character

string and one or more candidate document words (i.e., a combination of the predicted words and the candidate document words).

[0079] The similarity determination module 135 is configured to compare predicted words of the predicted character string to document words of the document text and determine a similarity measure of the predicted words and document word pairs. The similarity measure is indicative of the similarity between the predicted word and the document word of the pair.

[0080] The similarity determination module 135 is configured to assess the determined similarity measure against a similarity threshold and responsive to the determined similarity measure meeting the similarity threshold, the system 150 determines the document word as forming part of a candidate document string deemed to be sufficiently similar to the generative model prediction. On the other hand, if the determined similarity measure does not meet the similarity threshold, the similarity determination module 135 is configured to determine the document word as being insufficiently similar, or sufficiently different to predicted word. In the latter case, the document word does not form part of the candidate document string deemed to be sufficiently similar to the generative model prediction.

[0081] In some embodiments, to qualify as sufficiently similar, the predicted word and the document word must be identical. In other words, the similarity threshold is that the words of the pair are the same. If the similarity determination module 135 determines that the predicted word corresponds with the document word, the system 150 determines that the document word forms part of the candidate document string deemed to be sufficiently similar to the generative model prediction, and for example, that the determined document attribute comprises the predicted word.

[0082] In some embodiment, sufficiently similarity may require the predicted words of the predicted character string to be similar to, but not necessarily to correspond with or match exactly, the document words. In some such embodiments, the similarity

threshold may be based on an exact match, and an edit distance calculation and/or a substring matching process.

[0083] To comply with the exact match threshold (being a first threshold), the predicted word must match the document word of the pair. Where this is found to be true, a first similarity measure is assigned to the pair. If the similarity measure of the pair does not meet the first threshold, the similarity measure of the pair is compared with a second threshold, being the edit distance threshold.

[0084] To comply with the edit distance threshold, the predicted word must be of a length greater than threshold length (e.g., 3 characters), and an edit distance value of the predicted word and document word pair must meet an edit distance threshold (for example, less than '2'). The edit distance calculation may be based on the Levenshtein edit-distance or the Damerau–Levenshtein edit-distance calculation, for example. The Levenshtein edit-distance may be defined as a number of edits required by character deletion, insertion or substitution, to transform one character string or word into the other. The Damerau–Levenshtein edit-distance may be defined as a number of edits required by character deletion, insertion, substitution or transposition (swapping of adjacent characters) to transform one character string or word into the other.

[0085] Where the similarity measure is found to meet the second threshold, a second similarity measure is assigned to the pair. If the similarity measure of the pair does not meet the second threshold, the similarity measure of the pair is compared with a third threshold, being the substring threshold.

[0086] To comply with the substring matching threshold, the predicted word must be a substring of the document word of the pair, and must be of a length greater than threshold length (e.g., 3 characters).

[0087] A substring of a string or word is a sequence of contiguous characters forming a proper subset of the characters of the string or word. A proper subset of a set X is a subset of set X that is not equal to set X; if Y is a proper subset of X, then all elements

of set Y are in set X, but set X includes at least one element that is not in set Y. In other words, the number of characters in a substring is less than the number of characters in the full string or word.

[0088] Where the similarity measure is found to meet the third threshold, a third similarity measure is assigned to the pair. If the similarity measure of the pair does not meet the third threshold, the similarity measure of the pair is determined as being insufficient. In this case, the document word is disregarded or discounted as a possible word for the part of the candidate document string.

[0089] The similarity determination module 135 determines a best match for each predicted word based on a highest of the similarity measures for the predicted word and document word pairs. For example, a first similarity measure may be considered a highest ranked similarity measure, with the second similarity measure being the second highest ranked similarity measure and the third similarity measure being the third highest ranked similarity measure. The best match documents word for each predicted word of the predicted character string may be collated or joined together in a candidate document character string.

[0090] The attribute output module 372 is configured to determine a document attribute prediction based on the determination of the similarity determination module 135. The attribute output module 372 is further configured to provide the document attribute prediction as an output.

[0091] The determined document attributes output by the attribute output module 372 may be used in many applications, including for example, for data or transaction reconciliation. In some embodiments, the attributes determined may be transmitted to or may be made accessible to an accounting system (not shown) or a system (not shown) for storing, manipulating and/or reconciling accounting data. The extracted information, such as a vendor or originator or issuer name or identity, may be used within the accounting system (not shown) to reconcile a transaction associated with the candidate document with one or more accounting records in the accounting system (not

shown). This may allow for efficient and accurate extraction, tracking and reconciliation of transactional data associated with documents by automatically extracting transaction information from the documents and making it available to an accounting system (not shown) for reconciliation. The embodiments may also allow the extraction of transaction information from documents associated with expenses by individuals in an organisation. The extracted information may be transmitted or made available to an expense claim tracking system to track, approve and process expenses by individuals in an organisation.

[0092] The memory 356 may comprise an entity record management module 376 configured to engage with and manage a business register or business register data structure (not shown). For example, the entity record management module 376 may be configured to populate and/or update information stored in the business register based on determined document attributes provided by the attribute output module 372. The business register or business register data structure may be local to the system 150 or may be stored in data store 340 or a graph structure (not shown). In some embodiments, the entity record management module 376 is deployed on a separate system remote to system 150 and configured to cooperate with system 150, in some embodiments, to allow the system 150, and for example, the attribute determination module 362, to populate and/or update information stored in the business register based on determined document attributes.

[0093] The memory 356 may comprise a transaction reconciliation module 374 for reconciling or facilitating the reconciling of financial transactions. For example, in some embodiments, a financial statement may be used by the transaction reconciliation module to reconcile financial transactions (such as bank statements) against accounting documents, such as invoices or credit notes, issued by an organisation. The transaction reconciliation module 374 is configured to provide financial or accounting documents or records to the attribute determination module 362 of the system 150 (for example, by way of an attribute determination request), and in response to receive determined document attribute(s) of the provided documents or record. The determined attribute(s)

may then be used by the transaction reconciliation module 374 to reconcile the associated transaction.

[0094] **Figure 4** is a process flow diagram of a method 400 of determining document attributes, according to some embodiments. The method 400 may be performed by the system 150 or components thereof.

[0095] At 405, the system 150, for example, the data handling module 358 of the attribute determination module 362, determines document text associated with a document for which document attribute(s) are desired or sought to be determined. The document text comprises a plurality of document words.

[0096] The system 150, for example, the data handling module 358, may receive a document or document text for processing/analysis or to otherwise determine one or more document attributes. In some embodiments, data handling module 358, working in combination with the communications interface 354 may receive a document or document text from client device 310, database 340 or financial institute systems 345, or any other suitable source. The data handling module 358 may be configured to perform one or more quality assurance processes, such as confirming that the received data is not corrupted, and/or that the data is being sent from a verified or otherwise secure source. The data handling module 358 may comprise or otherwise be configured to interact with one or more API to facilitate the secure transfer of data to/from the attribute determination system 150.

[0097] The system 150 extracts, or otherwise determines, the text to be processed for determining the one or more document attributes. Where a raw document includes a mix of text and images, for example, an image-heavy Portable Document Format (PDF), the pre-processing module 364 performs optical character recognition (OCR) to determine the document text. The pre-processing module 364 may be configured to determine whether an OCR process is required or not. For example, responsive to the pre-processing module 364 determining that the raw document includes a threshold proportion of embedded images including text, the pre-processing module 364

performs an OCR process to extract the text from the images to thereby determine the document text.

[0098] The system 150, for example, the tokenization module 366 of the attribute determination module 362, converts or tokenises the document text into one or more tokens. In some embodiments, the text may be tokenised based on one or more criterion. The tokenisation criteria may be based on one or more of a symbol, element or other characteristic of the document text. For example, the text may be broken into tokens based on text symbols such as ",", ".", and/or "-". In some embodiments, the text may be broken into tokens based on spaces between blocks of text or strings, or otherwise white space between text. In some embodiments, the document text is tokenised into sub-words or word pieces using a pre-trained Hugging face tokenizer (t5-base). For example, the tokenization module 366 would tokenize the text "home depot" into tokens "home" "de" and "pot".

[0099] At 410, the system 150, for example, the numerical representation model 368 of the attribute determination module 362, determines a numerical representation of the document token(s).

[0100] At 415, the system 150, for example, the numerical representation model 368, provides the numerical representation of the token(s) to the generative model 125.

[0101] At 420, the attribute determination system 150, for example, the generative model 125 of the attribute determination system 150, determines a generative model prediction of the document attribute based on the numerical representation of the token(s). The generative model prediction comprises a predicted character string. The predicted character string comprises one or more predicted words.

[0102] The predicted character string is indicative of a predicted document attribute. For example, the predicted character string may be indicative of the entity that generated or issued the document, or the entity to whom the document was issued, or may be an entity named or listed within the document.

[0103] At 425, the attribute determination system 150, for example, the similarity determination module 135, determines a candidate document character string of the document text that is sufficiently similar to the predicted character string. The candidate document character string comprises one or more of the plurality of document words.

[0104] For example, the system 150 may, for each predicted word, determine a suitability measure indicative of the similarity of the predicted word to one or more or all of the document words of the document text.

The system 150 may be configured to parse the document text comparing the predicted word to each document word to determine suitability measure for the pairs of words. In some embodiments, responsive to the system 150, for example, the similarity determination module 135, determining that the suitability measure for the pair of words is sufficiently similar, the system 150 ceases parsing the document text for that predicted word, and returns an indication that the predicted word and the document word are substantially similar. In other embodiments, the similarity determination module 135 may determine a similarity measure for all possible predicted word and document word pairs and may then select a best match document word, such as the document word with the highest similarity measure. The document words of the sufficiently similar predicted word and document word pairs, or the best match document words are selected to form the candidate document character string deemed sufficiently similar to the predicted character string. The system 150 assesses the determined similarity measure against a similarity threshold and responsive to the determined similarity measure meeting the similarity threshold, the system 150 determines the document word as being sufficiently similar to the predicted word of the generative model prediction (i.e., the predicted character string). On the other hand, if the determined similarity measure does not meet the similarity threshold, the system 150 determines the document word as being insufficiently similar, or sufficiently different, to the predicted word of the generative model prediction.

[0105] In some embodiments, the similarity measure is determined as being a first similarity measure, a second similarity measure or a third similarity measure, wherein the first similarity measure is indicative a greater similarity than the second and third similarity measure and the second similarity measure is indicative a greater similarity than the third similarity measure.

[0106] For each predicted word of the predicted character string, the similarity determination module 135 determines if pairs of the predicted word and a document word meet a first similarity threshold. The first similarity threshold is determinative of whether or not the predicted word is the same as the document word of the pair. Accordingly, only predicted word and document word pairs where the predicted word is the same as the document word comply with the first threshold. The similarity measure for these pairs of words is a first similarity measure.

[0107] For each similarity measure of a predicted word and a document word pairthat does not comply with the first threshold, the similarity determination module 135 determines if the pair meet a second threshold. The second threshold is that the predicted word meets a threshold length (e.g., 3 characters long), and a threshold edit distance (e.g. 1). Accordingly, only predicted word and document word pairs where the predicted words meet the threshold length and edit distance threshold comply with the second threshold. The similarity measure for these pairs of words is a second similarity measure.

For each similarity measure of a predicted word and a document word pair that does not comply with the second threshold, the similarity determination module 135 determines if the pair meet a third threshold. The third threshold is that the predicted word meets a threshold length (e.g., 3 characters long), and is a substring of a document word of the candidate document character string. Accordingly, only predicted word and document word pairs where the predicted words meet the threshold length and are substrings of the document word comply with the third threshold. The similarity measure for these pairs of words is a third similarity measure.

[0108] At 430, the system 150, for example the attribute determination module 362, determines the document attribute based on the predicted character string and/or the candidate document character string.

[0109] For example, the determined document attribute may comprise, or contain the predicted character string; the candidate document character string; a combination of words from the predicted character string and the candidate document string (i.e., one or one or more predicted words of the predicted character string and one or more document words of the candidate document character string).

[0110] At 435, the system 150, for example the output module 372 of the attribute determination module 362, outputs the determined document attribute.

[0111] As mentioned above, the attribute determination module 362 may output the determined document attribute for use by one or more other modules or processes, being executed by system 150 or elsewhere. In one example, the determined document attribute may be provided to the transaction reconciliation module 374 for aiding in the reconciliation of transactions. In one example, the determined document attribute may be provided to the entity record management module 376 of populating or updating records, such as a business register or contact list. In one example, the determined document attribute may be provided to the client device 310 for presentation to a user via device I/O 320.

[0112] Determining a version of the generative model 125 comprised training a default configured T5 model sourced from the Huggingface library. The default configured T5 model comprised an input sequence length of 512 tokens, an output sequence length of 64 tokens, a batch size of 8, and a cased vocabulary of 32,128. The default configured T5 model further comprised an adam optimizer and was also fully finetuned. The default configured T5 model was trained on the sequence-to-sequence task of directly predicting the supplier name given only the text input from a document.

[0113] During testing and development of some of the described embodiments, a version of the generative model 125 was determined by training a default configured T5 model sourced from the Huggingface library. The default configured T5 model comprised an input sequence length of 512 tokens, an output sequence length of 64 tokens, a batch size of 8, and a cased vocabulary of 30,000. The default configured T5 model further comprised an adam optimizer and was also fully finetuned. The default configured T5 model was trained on the sequence-to-sequence task of directly predicting the supplier name given only the text input from a document.

[0114] The default configured T5 model was trained using a random selection of documents dated from 1 January 2022 to 31 December 2022. The documents were sourced from a Hubdoc, Bills and Expenses repositories and included documents of type invoice, receipt, credit note, bill, each of which having a verified supplier name.

[0115] Once the trained T5 model (generative model 125) was fully trained using the downstream training task, a post processing process was introduced to limit the predictions of the generative model 125 to only strings that were present in the document text. A harsh (strict match) post processing process was implemented comprising only accepting predictions that occur exactly in the document text, as described above. Further, a fuzzy post-processing process, as described above was implemented.

[0116] The above-described training process and subsequently trained model, combined with the above described strict/exact and fuzzy post processing steps produced the following results (Table I and Table II) when determining a supplier of a document.

Exact match accuracy	Lower match accuracy	Substring match accuracy	Lower substring match accuracy	Average damerua levenshtein distance	Average jaro score	support
Baseline T5 – No post processing						
0.44642	0.49074	0.66596	0.76268	6.51712	0.837384	100000
Baseline T5 – Harsh (strict match) post processing						
0.4353	0.47837	0.64406	0.73433	6.89424	0.785527	100000

Baseline T5 – Fuzzy post-processing V1						
0.44355	0.48948	0.66458	0.76474	6.50554	0.829935	100000
Baseline T5 – Fuzzy post-processing V2						
0.44451	0.49063	0.66637	0.76718	6.50407	0.830074	100000

Table I

Model	**Exact match accuracy**	**Substring match accuracy**	**Lower substring match accuracy**
T5 baseline	44.6%	66.6%	76.3%
T5 + strict match	43.5%	64.4%	73.3%
T5 + fuzzy match	44.5%	66.6%	76.7%

Table II

[0117] In the above table the metrics are: exact match accuracy: the model prediction matches exactly the supplier name the user submits; substring match accuracy: the model prediction matches exactly the supplier name the user submits, after both the prediction and user submission are cast to lowercase; and lower substring match accuracy: the model prediction is a substring of the user submission, after both are cast to lowercase, OR the user submission is a substring of the model prediction, after both are cast to lowercase.

[0118] It will be appreciated by persons skilled in the art that numerous variations and/or modifications may be made to the above-described embodiments, without departing from the broad general scope of the present disclosure. The present embodiments are, therefore, to be considered in all respects as illustrative and not restrictive.

CLAIMS:

1. A method comprising:

determining document text comprising a plurality of document words, the document text being associated with a document for which a document attribute is sought to be determined;

determining a numerical representation of the document text;

providing the numerical representation to a generative model to determine a generative model prediction of the document attribute, the generative model prediction comprising a predicted character string, the predicted character string comprising one or more predicted words;

determining a candidate document character string of the document text that is sufficiently similar to the predicted character string, the candidate document character string comprising one or more of the plurality of document words;

determining the document attribute based on the predicted character string and/or the candidate document character string; and

outputting the determined document attribute.

2. The method of claim 1, wherein the determined document attribute comprises:

(i) the predicted character string;

(ii) the candidate document character string; or

(iii) one or more predicted words of the predicted character string and one or more document words of the candidate document character string.

3. The method of claim 1 or 2, wherein determining the candidate document character string comprises:

for each predicted word and document word pair, determining a similarity measure indicative of the similarity of the predicted word and document word of the pair;

for each predicted word, determining the predicted word and document word pair with a highest similarity measure; and

selecting the document word of the predicted word and document word pair as a document word of the candidate document character string.

4. The method of claim 3, wherein determining the similarity measure for a predicted word and document word pair comprises:

determining whether the predicted word and document word pair meets a first, second or third similarity threshold, wherein the first threshold is that the predicted word corresponds exactly with the document word, the second threshold is that the predicted word is of a threshold length and is within a threshold edit distance of the document word, and the third threshold is that the predicted word is of a threshold length and is a substring of the document word; and

responsive to determining that the predicted word and document word pair meets the first, second or third threshold, determining the similarity measure as a first similarity measure, a second similarity measure, or a third similarity measure, respectively.

5. The method of claim 4, wherein the first similarity measure is indicative of a greater level of similarity than the second similarity measure and the third similarity measure, and the second similarity measure is indicative of a greater level of similarity than the third similarity measure.

6. The method of claim 3, wherein determining the similarity measure for a predicted word and document word pair comprises:

determining whether the predicted word corresponds to the document word of the predicted word and document word pair; and

responsive to determining that the predicted word corresponds to the document word, determining the predicted word and document word pair as the predicted word and document word pair with the highest similarity measure for the predicted word.

7. The method of claim 6, wherein determining that the predicted word corresponds to the document word comprises determining that the predicted word and document word pair meet a first similarity threshold, and assigning the predicted word and document word pair a first similarity measure.

8. The method of any of claims 1 to 3, 6 or 7, wherein determining the similarity measure for a predicted word and document word pair comprises:

determining whether the predicted word and document word pair meets a second similarity threshold, wherein the second similarity threshold is that the predicted word is of a threshold length and is within a threshold edit distance of the document word; and

responsive to the predicted word and document word pair meeting the second similarity threshold, assigning the predicted word and document word pair a second similarity measure.

9. The method of any of claims 1 to 4, or 6 to 8 , wherein determining the similarity measure for a predicted word and document word pair comprises:

determining whether the predicted word and document word pair meets a third similarity threshold, wherein the third similarity threshold is that the predicted word is of a threshold length and is a substring of the document word; and

responsive to the predicted word and document word pair meeting the third similarity threshold, assigning the predicted word and document word pair a third similarity measure.

10. The method of any one of the preceding claims, wherein responsive to determining that each of the predicted words of the predicted character string corresponds to a respective document word of the candidate character document string, determining the predicted character string as the document attribute.

11. The method of claim 7, or any one of claims 8 to 10 when directly or indirectly dependent on claim 7, wherein responsive to the highest similarity measure

for predicted word and document word pair being the first similarity measure, determining the document attribute to comprise the predicted word of the predicted word and document word pair.

12. The method of claim 8 or any one of claims 9 to 11 when directly or indirectly dependent on claim 8, wherein responsive to the highest similarity measure for predicted word and document word pair being the second similarity measure, determining the document attribute to comprise the predicted word of predicted word and document word pair.

13. The method of claim 9 or any one of claims 10 to 12 when directly or indirectly dependent on claim 9, wherein responsive to the highest similarity measure for predicted word and document word pair being the third similarity measure, determining the document attribute to comprise the document word of predicted word and document word pair.

14. The method of any one of the preceding claims, wherein determining document text associated with a document for which a document attribute is sought to be determined comprises:

removing any non-alphanumeric characters and/or converting all text to lower case.

15. The method of any one of the preceding claims, wherein determining document text comprises tokenizing the document text into one or more document tokens, and wherein determining a numerical representation of the document text comprises determining a numerical representation of the one or more document tokens.

16. The method of any one of the preceding claims, further comprising:

providing the determined document attribute to a reconciliation module; and

reconciling a transaction associated with the document using the determined document attribute.

17. The method of any one of the preceding claims, further comprising:

providing the determined document attribute to an entity attribute management module; and

(i) creating a new entity profile in a database using the determined document attribute, or (ii) updating an existing entity profile in the database using the determined document attribute.

18. A system comprising:

one or more processors; and

memory comprising computer executable instructions, which when executed by the one or more processors, cause the system to perform the method of any one of claims 1 to 17.

19. A computer-readable storage medium storing instructions that, when executed by a computer, cause the computer to perform the method of any one of claims 1 to 17.

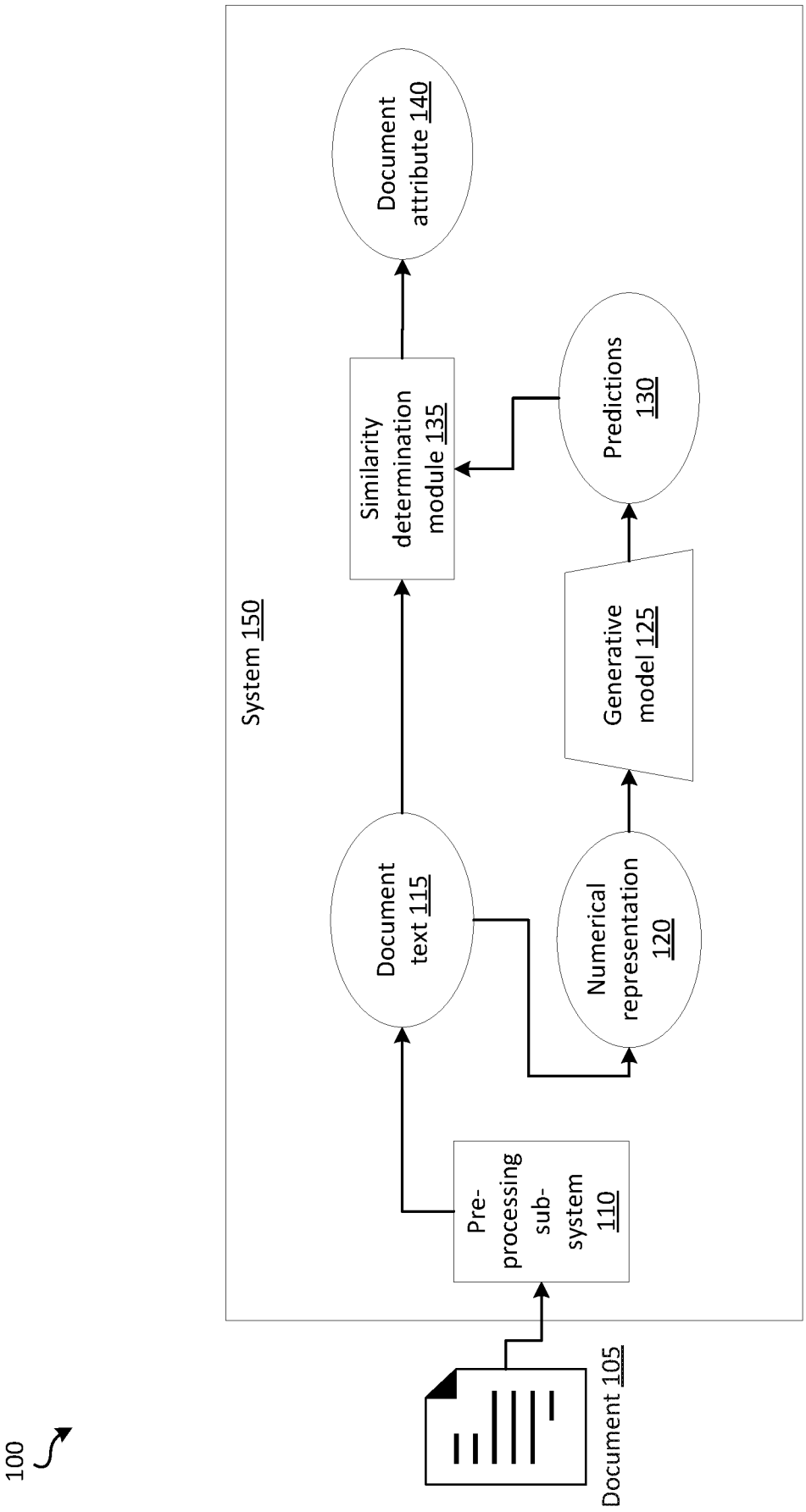
100
Document 105
System 150
Pre-processing sub-system 110
Document text 115
Numerical representation 120
Generative model 125
Predictions 130
Similarity determination module 135
Document attribute 140

Fig. 1

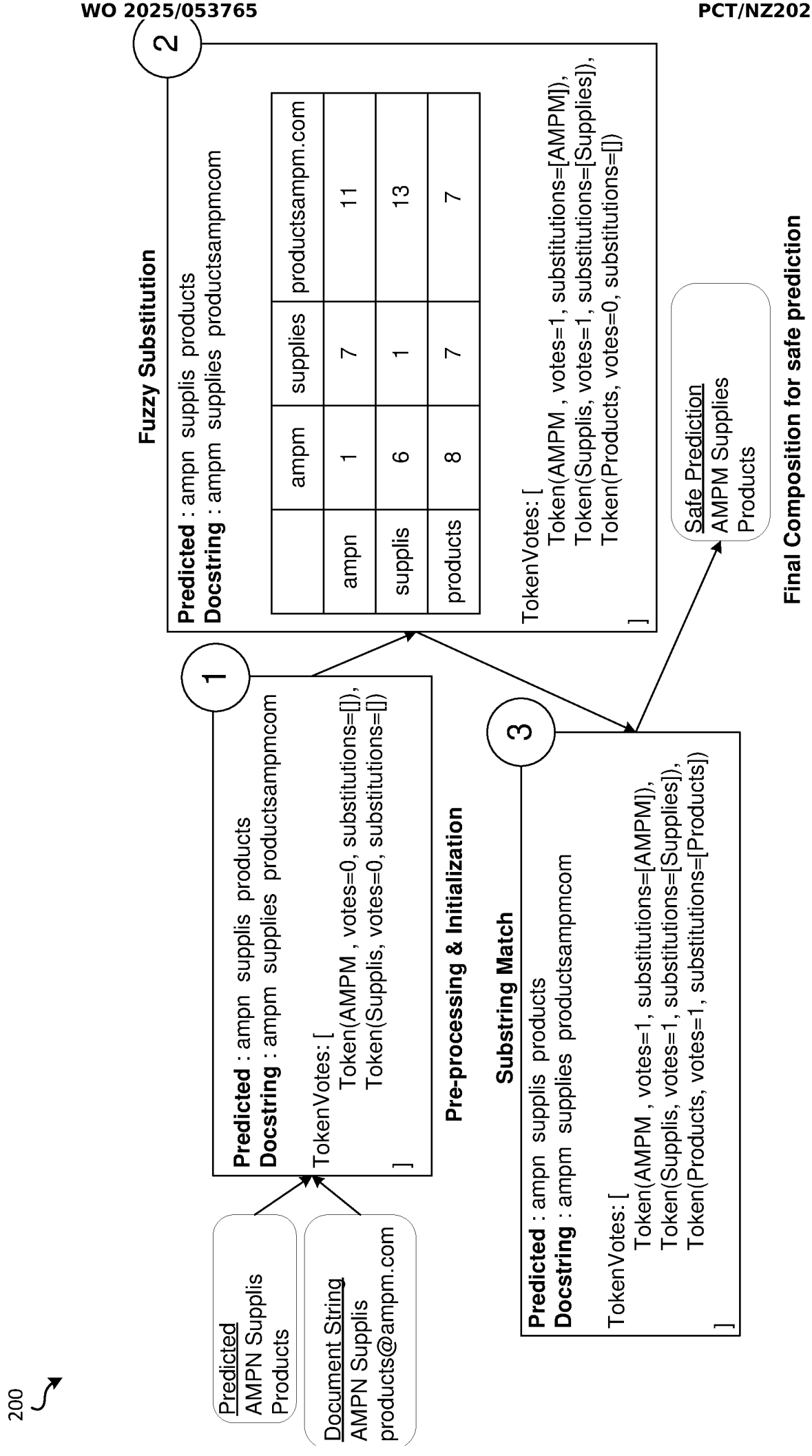
200
Predicted
AMPN Supplis
Products
Document String
AMPN Supplis
products@ampm.com
1
Predicted : ampn supplis products
Docstring : ampm supplies productsampmcom
TokenVotes: [
Token(AMPM , votes=0, substitutions=[]),
Token(Supplis, votes=0, substitutions=[])
]
Pre-processing & Initialization
2
Fuzzy Substitution
Predicted : ampn supplis products
Docstring : ampm supplies productsampmcom
ampm
supplies
productsampm.com
ampn 1 7 11
supplis 6 1 13
products 8 7 7
TokenVotes: [
Token(AMPM , votes=1, substitutions=[AMPM]),
Token(Supplis, votes=1, substitutions=[Supplies]),
Token(Products, votes=0, substitutions=[])
]
3
Substring Match
Predicted : ampn supplis products
Docstring : ampm supplies productsampmcom
TokenVotes: [
Token(AMPM , votes=1, substitutions=[AMPM]),
Token(Supplis, votes=1, substitutions=[Supplies]),
Token(Products, votes=1, substitutions=[Products])
]
Safe Prediction
AMPM Supplies
Products
Final Composition for safe prediction

Fig. 2

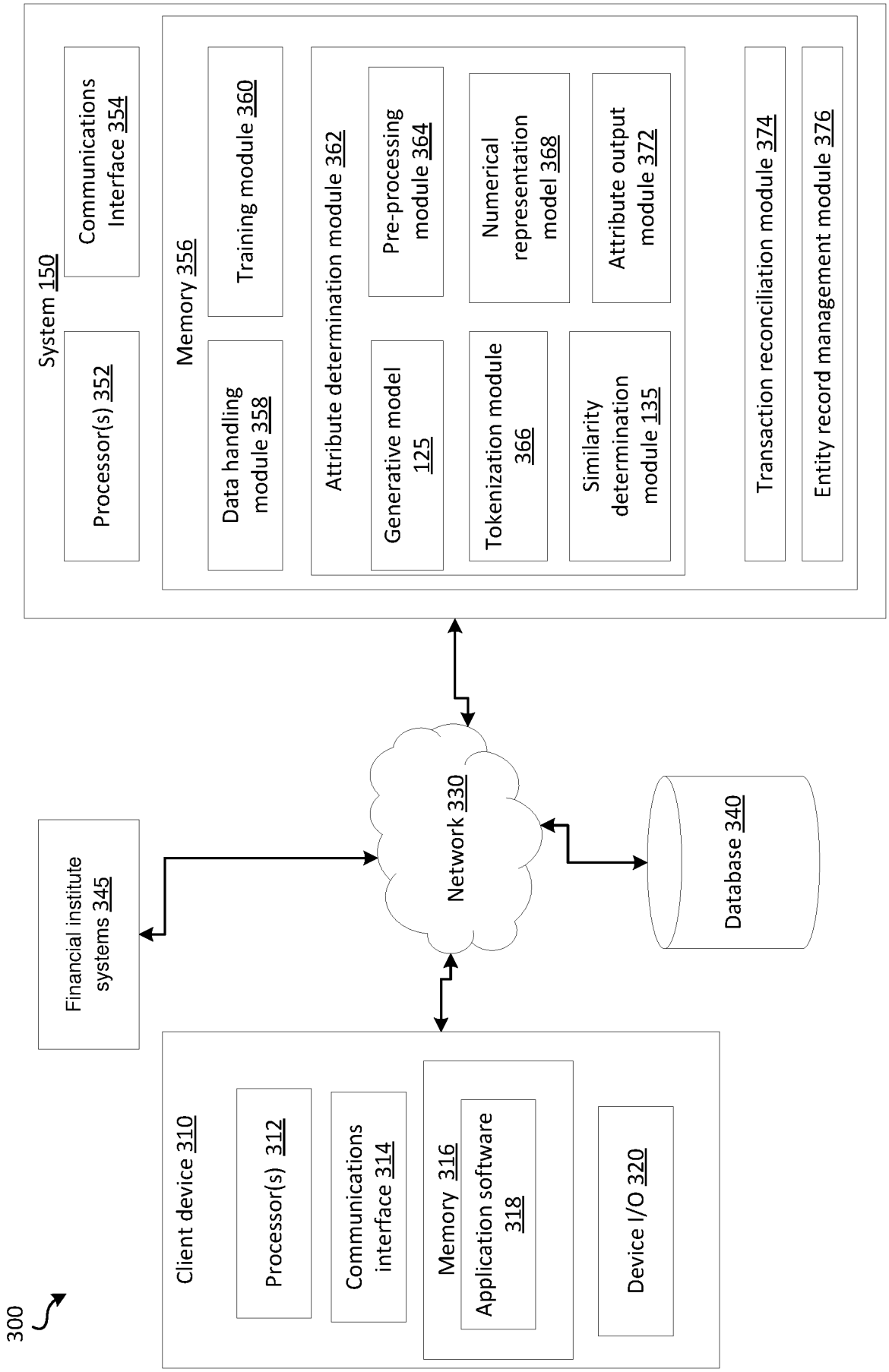
300
Client device 310
Processor(s) 312
Communications interface 314
Memory 316
Application software 318
Device I/O 320
Financial institute systems 345
Network 330
Database 340
System 150
Processor(s) 352
Communications Interface 354
Memory 356
Data handling module 358
Training module 360
Attribute determination module 362
Generative model 125
Pre-processing module 364
Tokenization module 366
Numerical representation model 368
Similarity determination module 135
Attribute output module 372
Transaction reconciliation module 374
Entity record management module 376

Fig. 3

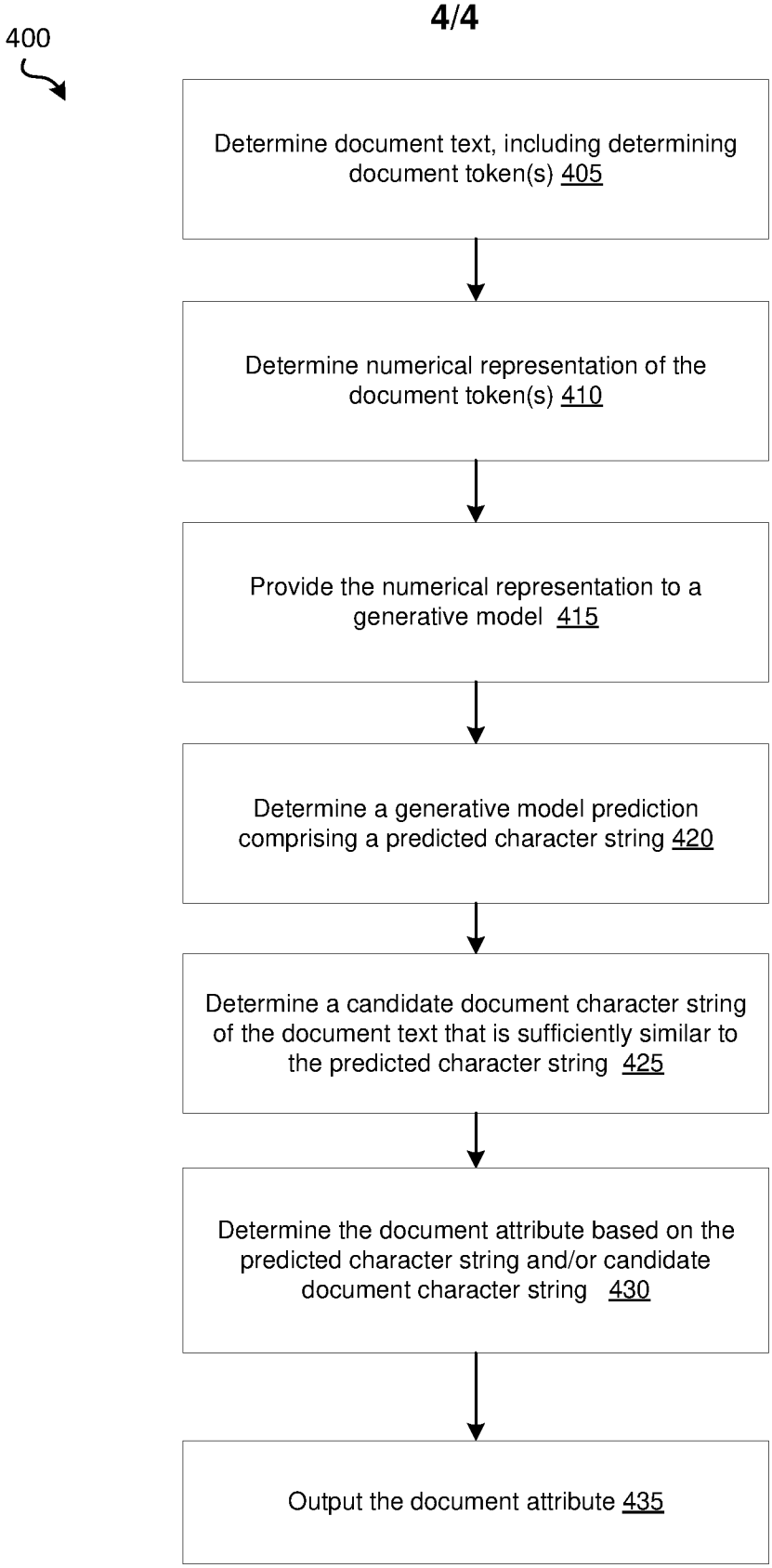
400
Determine document text, including determining document token(s) 405
Determine numerical representation of the document token(s) 410
Provide the numerical representation to a generative model 415
Determine a generative model prediction comprising a predicted character string 420
Determine a candidate document character string of the document text that is sufficiently similar to the predicted character string 425
Determine the document attribute based on the predicted character string and/or candidate document character string 430
Output the document attribute 435

Fig. 4

INTERNATIONAL SEARCH REPORT

International application No.
PCT/NZ2024/050104

A. CLASSIFICATION OF SUBJECT MATTER

IPC: ***G06F 16/20*** (2023.01); ***G06F 16/30*** (2023.01); ***G06F 40/284*** (2023.01); ***G06N 3/08*** (2023.01); ***G06N 20/00*** (2023.01); *G06F 16/00* (2023.01)

CPC: ***G06F 16/20; G06F 16/30; G06F 40/284; G06N 3/08; G06N 20/00;*** *G06F 16/00*

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)

See Search History Document

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

See Search History Document

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

See Search History Document

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2019/0339824 A1 (Adhark, Inc.) 07 November 2019 (07.11.2019) entire document, especially Para [0004],[0011],[0015],[0045],[0047],[0060],[0063],[0066]-[0067],[0073],[0075],[0078],[0093]	1-7

☐ 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
14 November 2024 (14.11.2024)	**11 December 2024 (11.12.2024)**

Name and mailing address of the ISA/US	Authorized officer
COMMISSIONER FOR PATENTS **MAIL STOP PCT, ATTN: ISA/US** **P.O. Box 1450** **Alexandria, VA 22313-1450** **UNITED STATES OF AMERICA**	**KARI RODRIQUEZ**
Facsimile No. **571-273-8300**	Telephone No. **PCT Help Desk: 571-272-4300**

INTERNATIONAL SEARCH REPORT

International application No.
PCT/NZ2024/050104

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☑ Claims Nos.: **8-19**
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).