March 2023

USE OF GRAPH DATABASES TO DETECT ILLEGITIMATE MONEY TRANSFERS

Partha Saha
Visa

Roger Huang
Visa

Hemant Deshpande
Visa

Sumit Lal
Visa

Sam Hamilton
Visa

See next page for additional authors

Inventor(s)
Partha Saha, Roger Huang, Hemant Deshpande, Sumit Lal, Sam Hamilton, and Suman Mukherjee

TITLE: "USE OF GRAPH DATABASES TO DETECT ILLEGITIMATE MONEY TRANSFERS"

VISA

<u>INVENTORS:</u>

PARTHA SAHA

ROGER HUANG

HEMANT DESHPANDE

SUMIT LAL

SAM HAMILTON

SUMAN MUKHERJEE

TECHNICAL FIELD

[0001] This disclosure relates generally to the field of application security. More particularly, the present disclosure relates to a system and method for usage of a graph database to detect illegitimate money transfers.

BACKGROUND

[0002] In current banking and other financial organizations, several payment flows including Business-to-Business payments take place on a day to day basis. However, these transaction need to be approved as the transaction might be illegitimate or fraudulent or illicit. Therefore, there is a need for a scalable "graph" database which can be cross checked as a transaction comes in to see if any party either directly or indirectly related to the transaction seems to be suspicious (i.e., flagged in one of the appropriate news databases). There is also a requirement to see if a transaction is part of a connected set of transactions that seems suspicious. Therefore, there is a need for an improved method for detecting fraudulent activity.

[0003] Therefore there is a need for an efficient way of solving one or more of the above mentioned problems.

SUMMARY

[0004] One embodiment of the invention includes a method. The method comprises: determining an entity identifier and a geographic location identifier; forming an article identifier using the entity identifier and the geographic location identifier; grouping articles pertaining to the entity identifier and the geographic location identifier; storing the articles in an article database indexed to the entity identifier and the geographic location identifier; identifying a suspicious entity in a graph database comprising nodes of articles and edges with associations between the articles; determining entities associated with the suspicious entity using the graph database; and retrieving articles from the article database associated with the suspicious entity and the determined entities to determine if the suspicious entity is engaged in fraudulent activity.

[0005] A better understanding of the nature and advantages of embodiments of the invention may be gained with reference to the following detailed description and accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] The features and characteristics of the present invention, as well as the methods of operation and functions of the related elements of structures and the combination of parts and economies of manufacture, will become more apparent upon consideration of the following description and the appended claims with reference to the accompanying drawings, all of which form a part of this specification, wherein like reference numerals designate corresponding parts in the various figures. It is to be expressly understood, however, that the drawings are for the purpose of illustration and description only and are not intended as a definition of the limits of the invention. As used in the specification, the singular form of "a," "an," and "the" include plural referents unless the context clearly dictates otherwise.

[0007] Additional advantages and details of non-limiting embodiments are explained in greater detail below with reference to the exemplary embodiments that are illustrated in the accompanying schematic figures, in which:

[0008] FIG. 1 shows a block diagram of a system according to embodiments.

[0009] FIG. 2 shows a block diagram of an article database and a graph database according to embodiments.

[0010] FIG. 3 is a block diagram of an exemplary computer system for implementing embodiments consistent with the present disclosure.

[0011] FIG. 4 depicts a flow diagram of an exemplary method for detecting illegitimate money transfers in accordance with the present invention.

DETAILED DESCRIPTION

[0012] In the present document, the word "exemplary" is used herein to mean "serving as an example, instance, or illustration." Any embodiment or implementation of the present subject matter described herein as "exemplary" is not necessarily to be construed as preferred or advantageous over other embodiments.

[0013] While the disclosure is susceptible to various modifications and alternative forms, specific embodiment thereof has been shown by way of example in the drawings and will be described in detail below. It should be understood, however that it is not intended to limit the disclosure to the particular forms disclosed, but on the contrary, the disclosure is to cover all modifications, equivalents, and alternative falling within the spirit and the scope of the disclosure.

[0014] The terms “comprises”, “comprising”, or any other variations thereof, are intended to cover a non-exclusive inclusion, such that a setup, device or method that comprises a list of components or steps does not include only those components or steps but may include other components or steps not expressly listed or inherent to such setup or device or method. In other words, one or more elements in a device or system or apparatus proceeded by “comprises… a” does not, without more constraints, preclude the existence of other elements or additional elements in the device or system or apparatus.

[0015] The terms "an embodiment", "embodiment", "embodiments", "the embodiment", "the embodiments", "one or more embodiments", "some embodiments", and "one embodiment" mean "one or more (but not all) embodiments of the invention(s)" unless expressly specified otherwise.

[0016] The terms "including", "comprising", “having” and variations thereof mean "including but not limited to", unless expressly specified otherwise.

[0017] As used herein, the terms “communication”, “communicate”, “post”, “sent”, “return” and “returned” may refer to the reception, receipt, transmission, transfer, provision, and/or the like of information (e.g., data, signals, messages, instructions, commands, and/or the like). For one unit (e.g., a device, a system, a component of a device or system, combinations thereof, and/or the like) to be in communication with another unit means that the one unit is able to directly or indirectly receive information from and/or transmit information to the other unit. This may refer to a direct or indirect connection (e.g., a direct communication connection, an indirect communication connection, and/or the like) that is wired and/or wireless in nature. Additionally, two units may be in communication with each other even though the information transmitted may be modified, processed, relayed, and/or routed between the first and second unit. For example, a first unit may be in communication with a second unit even though the first unit passively receives information and does not actively transmit information to the second unit. As another example, a first unit may be in communication with a second unit if at least one intermediary unit (e.g., a third unit located between the first unit and the second unit) processes information received from the first unit and communicates the processed information to the second unit. In some non-limiting embodiments, a message may refer to a network packet (e.g., a data packet and/or the like) that includes data. It will be appreciated that numerous other arrangements are possible.

[0018] As used herein, the term “computer” or “computer system” may refer to one or more electronic devices that are configured to directly or indirectly communicate with or over one or more networks. A computer system may be a mobile or portable computing device, a desktop computer, a server, mobile phones (e.g., cellular phones), PDAs, tablet computers, net books, laptop computers, personal music players, hand-held specialized readers, wearable devices (e.g., watches), vehicles (e.g., cars) and/or the like. Furthermore, the term “computer” may refer to any computing device that includes the necessary components to receive, process, and output data, and normally includes a display, a processor, a memory, an input device, and a network interface. A “computer system” may include one or more computing devices or computers. An “application” or “Application Program Interface” (API) refers to computer code or other data sorted on a computer-readable medium that may be executed by a processor to facilitate the interaction between software components, such as a client-side front-end and/or server-side back-end for receiving data from the client. An “interface” refers to a generated display, such as one or more graphical user interfaces (GUIs) with which a user may interact, either directly or indirectly (e.g., through a keyboard, mouse, touchscreen, etc.). Further, multiple computers, e.g., servers, directly or indirectly communicating in the network environment may constitute a “system” or a “computing system”.

[0019] As used herein, the term "user" may include an individual. In some embodiments, a user may be associated with one or more personal accounts and/or devices.

[0020] As used herein, the term "processor" may refer to any suitable data computation device or devices. A processor may comprise one or more microprocessors working together to accomplish a desired function. The processor may include CPU comprises at least one high-speed data processor adequate to execute program components for executing user and/or system-generated requests. The CPU may be a microprocessor such as AMD's Athlon, Duron and/or Opteron; IBM and/or Motorola's PowerPC; IBM's and Sony's Cell processor; Intel's Celeron, Itanium, Pentium, Xeon, and/or XScale; and/or the like processor(s).

[0021] As used herein, the term "memory" may be any suitable device or devices that can store electronic data. A suitable memory may comprise a non-transitory computer readable medium that stores instructions that can be executed by a processor to implement a desired method. Examples of memories may comprise one or more memory chips, disk drives, etc. Such memories may operate using any suitable electrical, optical, and/or magnetic mode of operation.

[0022] It will be apparent that systems and/or methods, described herein, can be implemented in different forms of hardware, software, or a combination of hardware and software. The actual specialized control hardware or software code used to implement these systems and/or methods is not limiting of the implementations. Thus, the operation and behavior of the systems and/or methods are described herein without reference to specific software code, it being understood that software and hardware can be designed to implement the systems and/or methods based on the description herein.

[0023] Prior to discussing embodiments of the disclosure, some terms can be described in further detail.

[0024] An “article” may include a piece of writing. In some embodiments, an article may be a news article, a magazine, a blog, etc.

[0025] An “article tag” may include a label of an article. In some embodiments, an article tag may include a geographic tag or an entity tag. Examples of a geographic tag can include a city name, a country name, a city name, etc. Examples of an entity tag can include name tags (e.g., a full name of an individual) and company tags (e.g., a name of a business). As an illustration, an article can include geographic tags such as “San Francisco,” “California,” “United States,” entity tags including name tags such as “John Doe,” and company tags such as “First Bank,” “Local Department Store,” and “Central Bank.”

[0026] FIG. 1 shows a block diagram of a system according to embodiments. The system may comprise an article source computer 100, and a server computer 102 operating an article database 200. In some embodiments, the server computer 102 may communicate with a plurality of article source computers 100_1, 100_2, …100_n. Further, the article source computer refers to one or more of computer systems that stores or has data assets, databases, data sources, web documents, web pages, websites, documents, news blogs, news feeds etc.

[0027] The article source computer 100 may provide one or more articles to the server computer 102. For example, the server computer 102 may receive an article from the article source computer 100 relating to John Doe living in San Francisco receiving a large amount of funds from a First Bank. The received article may have a geographic tag such as “San Francisco” and entity tags such as “John Doe” and “First Bank.”

[0028] The server computer 102 may then determine a geographic identifier from the geographic tag. The generation of the geographic identifier may include formatting of the geographic tag, such as standardizing the format of the geographic location (e.g., removing spaces, lowercase, etc.).

[0029] The server computer 102 may additionally determine entity identifiers using the entity tags. In some embodiments, the server computer 102 may generate a person identifier using a full name in the article tags (e.g., “John Doe”), and generate a company identifier using a name of a company in the article tags (e.g., “First Bank”). A person identifier and a company identifier may both be examples of an entity identifier. The server computer 102 may use the geographic identifier(s) and the entity identifier(s) to form an article identifier (e.g., a semantic identifier) for the article. The article identifier can be a concatenation of the geographic identifier(s) and the entity identifier(s), such as “SanFranciscoJohnDoeFirstBank.” A similar process can be performed for each article received to generate an article identifier for each article. Each article may then be stored in the article database 200 and may be indexed by the geographic location identifier(s) and the entity identifier(s).

[0030] Upon receiving a new article, the server computer 102 generates an article identifier for the new article, such as “SanFranciscoBobSmithFirstBank” and compares the article identifier of the new article to other article identifiers in the article database 200, or to article identifiers in the article database 200 that have common geographic location identifiers and/or entity identifiers. The server computer 102 may group the new article identifier and the old article identifier if the two are sufficiently similar. In some embodiments, the Levenshtein distance between the two article identifiers may be used to determine if they are similar. For example, if the Levenshtein distance is within 33%, the two article identifiers may be grouped together. In some embodiments, a latent semantic analysis or latent Dirichlet allocation algorithm may be used to form article groups. The article identifier “SanFranciscoJohnDoeFirstBank” and the article identifier “SanFranciscoBobSmithFirstBank” may form an article group. The grouping of articles can thus be generated using semantic connections formed between article identifiers through the common entity identifiers and/or geographic location identifiers contained in the article identifiers. By forming article groups and storing the articles themselves, it is possible to predict the impact quickly of “breaking news” on a financial system. A faster fraud identification time can allow more time to respond effectively.

[0031] An exemplary data model can thus include an entity index (e.g., an entity such as an individual or an organization/company), a geographic index (e.g., a location such as a city, state, country, zip code, etc.), news details (e.g., an article identifier that points to an article), news grouping (e.g., an article group that connects articles which have semantic connections through common entities and/or geographic locations), a news group identifier (e.g., an identifier that identifies a news/article group), and semantic connections between the articles to form article groups (e.g., semantic connections determined using the Levenshtein distance of the article identifiers (semantic identifier) that is formed by concatenating geographic location identifiers with entity identifiers).

[0032] In another embodiment, one or more of statistical algorithms, or Artificial Intelligence (AI) based algorithms, pattern matching algorithms etc. may be used to arrive at the above mentioned results.

[0033] FIG. 2 shows a diagram of an article database 200 and a graph database 210 according to embodiments. The article database 200 can store a plurality of articles, such as the ones described above. A first article 202, a second article 204, a third article 206, and a fourth article 208 may be received by the article source computer 100 of FIG. 1.

[0034] A plurality of nodes (e.g., key 212, key 214, ..., key 228) is shown in the graph database 210, where each node may correspond to an article identifier, or a “key.” The nodes include key 218, key 220, key 222, and key 228 can form an article group as shown by the shaded region. For example, the key 218 may be “SanFranciscoSecondBankFirstBank,” the key 220 may be “SanFranciscoSeattleFirstBank,” the key 222 may be “SanFranciscoJohnDoeFirstBank,” and the key 228 may be “SanFranciscoBobSmithFirstBank.” In the graph database 210, the edges between nodes may be typed and have properties. For example, an edge may represent a semantic connection between two nodes, and identify a relationship exists between the two nodes (e.g., “SanFranciscoJohnDoeFirstBank” and “SanFranciscoBobSmithFirstBank” can have an edge indicating they are similar). In some embodiments, the nodes may be typed and can include properties. Examples of types and properties of a node can include an article source (e.g., news article originating from Newspaper 1, blog post originating from blog website). The graph database 210 can be used to search for paths between nodes that match patterns, which is difficult to do efficiently with other technologies.

[0035] An investigation prompt 230 may be received by the server computer 102. The investigation prompt 230 may be received from an external computer that can request the server computer 102 to retrieve articles relating to an entity. For example, an entity such as a payment processing organization may monitor account money transfers between banks and may observe a suspicious activity associated with a particular account at one of the banks. The account may be held by transaction between a first bank and a second bank. For example, the payment processing organization may observe a transaction for a very large amount of money (e.g., $1,000,000) that is transferred from Bob Smith using the first bank operating in San Francisco to Joe Miller using the second bank operating in Atlanta. This is suspicious, because large amounts of money have not been transferred from Bob Smith's account in the past. In some embodiments, the investigation prompt may include further details of the suspicious transaction.

[0036] After receiving the investigation prompt 230, the server computer 102 may identify one or more suspicious entities in the graph database 210. For example, the server computer 102 may use the geographic location identifiers of San Francisco and Atlanta and the entity identifiers including Bob Smith, Joe Miller, first bank, and second bank to identify keys in the graph database 210. The suspicious entities may correspond to these entity identifiers. In embodiments, the elements of the identified key would correspond to the geographic location identifiers and/or the entity identifiers associated with the suspicious activity. In this example, the keys identified would be key 228 because it contains San Francisco, Bob Smith and first bank. The other key that would be identified include 218 with includes San Francisco, first bank, and second bank. From the identified keys 218 and 228, other related nodes/keys 220, 222 may be identified as being related to keys 218 and 228. This may be because of the similarities of the newly identified keys 220, 222 to the earlier identified keys 218, 228.

[0037] Using the newly identified keys 220, 222, additional entities associated with the suspicious entity can be determined. For example, the entity John Doe and the location Seattle can be identified by the keys 220, 222.

[0038] The server computer 102 may then retrieve articles from the article database that are associated with the suspicious entity and the determined entities. For example, the server computer

102 may use the suspicious entity identifiers Bob Smith, first bank, Joe Miller, and second bank and the determined entity identifier John Doe, in combination with the associated geographic identifiers, to access the article database and retrieve for articles related to the entities. For example, in FIG. 2, the second article 204 is related to the key 222, and the fourth article 208 is related to the key 228 and the key 222. The second article 204 and the fourth article 208 may then be transmitted to the external computer to determine if the suspicious entity is engaged in fraudulent activity. Suspicious activity monitoring techniques can be employed by the external computer to determine if the suspicious entity is engaged in fraudulent activity.

[0039] In some embodiments, an external computer (e.g., a payment processing network) may generate the investigation prompt 230 after onboarding an entity. As a part of the onboarding process, KYC data including an entity name, geographic locations associated with the entity, and a list of companies they are associated with can be received. The KYC (know your customer) data can be used to generate an entity identifier and a geographic location identifier, which can be used to determine which entities in the graph database 210 are related to the entity to be onboarded. The investigation prompt 230 can be a request to search the graph database 210 for articles relating to the entity to be onboarded.

[0040] Any of the software components or functions described in this application may be implemented as software code to be executed by a processor using any suitable computer language such as, for example, Java, C, C++, C#, Objective-C, Swift, or scripting language such as Perl or Python using, for example, conventional or object-oriented techniques. The software code may be stored as a series of instructions or commands on a computer readable medium for storage and/or transmission, suitable media include random access memory (RAM), a read only memory (ROM), a magnetic medium such as a hard-drive or a floppy disk, or an optical medium such as a compact disk (CD) or DVD (digital versatile disk), flash memory, and the like. The computer readable medium may be any combination of such storage or transmission devices.

[0041] Such programs may also be encoded and transmitted using carrier signals adapted for transmission via wired, optical, and/or wireless networks conforming to a variety of protocols, including the Internet. As such, a computer readable medium according to an embodiment of the present invention may be created using a data signal encoded with such programs. Computer readable media encoded with the program code may be packaged with a compatible device or provided separately from other devices (e.g., via Internet download). Any such computer readable medium may reside on or within a single computer product (e.g. a hard drive, a CD, or an entire computer system), and may be present on or within different computer products within a system or network. A computer system may include a monitor, printer, or other suitable display for providing any of the results mentioned herein to a user.

[0042] FIG. 3 is a block diagram of an exemplary computer system for implementing embodiments consistent with the present disclosure.

[0043] In some embodiments, FIG. 3 illustrates a block diagram of an exemplary computer system 300 for implementing embodiments consistent with the present disclosure. The processor 302 may include at least one data processor for executing program components for executing user or system-generated business processes. A user may include a person, a person using a device such as those included in this disclosure, or such a device itself. The processor 302 may include specialized processing units such as integrated system (bus) controllers, memory management control units, floating point units, graphics processing units, digital signal processing units, etc.

[0044] The processor 302 may be disposed in communication with input devices 311 and output devices 312 via I/O interface 301. The input devices 311 may be devices such as, without limitation to, keyboard, mouse, touch screen, sensors, microphones, scanners, camera, finger print scanner etc. The output devices 312 may be devices such as, without limitation to, speaker, electronic screen, etc. The I/O interface 301 may employ communication protocols/methods such as, without limitation, audio, analog, digital, stereo, IEEE-1393, serial bus, Universal Serial Bus (USB), infrared, PS/2, BNC, coaxial, component, composite, Digital Visual Interface (DVI), high-definition multimedia interface (HDMI), Radio Frequency (RF) antennas, S-Video, Video Graphics Array (VGA), IEEE 802.n /b/g/n/x, Bluetooth, cellular (e.g., Code-Division Multiple Access (CDMA), High-Speed Packet Access (HSPA+), Global System For Mobile Communications (GSM), Long-Term Evolution (LTE), WiMax, or the like), etc.

[0045] Using the I/O interface 301, the computer system 300 may communicate with the input devices 311 and the output devices 312.

[0046] In some embodiments, the processor 302 may be disposed in communication with a communication network 309 via a network interface 303. The network interface 303 may communicate with the communication network 309. The network interface 303 may employ connection protocols including, without limitation, direct connect, Ethernet (e.g., twisted pair 10/100/1000 Base T), Transmission Control Protocol/Internet Protocol (TCP/IP), token ring, IEEE 802.11a/b/g/n/x, etc. The communication network 309 can be implemented as one of the different types of networks, such as intranet or Local Area Network (LAN), Closed Area Network (CAN) and such. The communication network 309 may either be a dedicated network or a shared network, which represents an association of the different types of networks that use a variety of protocols, for example, Hypertext Transfer Protocol (HTTP), CAN Protocol, Transmission Control Protocol/Internet Protocol (TCP/IP), Wireless Application Protocol (WAP), etc., to communicate with each other. Further, the communication network 309 may include a variety of network devices, including routers, bridges, servers, computing devices, storage devices, etc. In some embodiments, the processor 302 may be disposed in communication with a memory 305 (e.g., RAM, ROM, etc. not shown in FIG.3) via a storage interface 303. The storage interface 303 may connect to memory 305 including, without limitation, memory drives, removable disc drives, etc., employing connection protocols such as Serial Advanced Technology Attachment (SATA), Integrated Drive Electronics (IDE), IEEE-1393, Universal Serial Bus (USB), fibre channel, Small Computer Systems Interface (SCSI), etc. The memory drives may further include a drum, magnetic disc drive, magneto-optical drive, optical drive, Redundant Array of Independent Discs (RAID), solid-state memory devices, solid-state drives, etc.

[0047] The memory 305 may store a collection of program or database components, including, without limitation, a user interface 306, an operating system 307, a web browser 308 etc. In some embodiments, the computer system 300 may store user/application data, such as the data, variables, records, etc. as described in this disclosure. Such databases may be implemented as fault-tolerant, relational, scalable, secure databases such as Oracle or Sybase.

[0048] The operating system 307 may facilitate resource management and operation of the computer system 200. Examples of operating systems include, without limitation, APPLE® MACINTOSH® OS X®, UNIX®, UNIX-like system distributions (E.G., BERKELEY SOFTWARE DISTRIBUTION® (BSD), FREEBSD®, NETBSD®, OPENBSD, etc.), LINUX® DISTRIBUTIONS (E.G., RED HAT®, UBUNTU®, KUBUNTU®, etc.), IBM®OS/2®, MICROSOFT® WINDOWS® (XP®, VISTA®/7/8, 10 etc.), APPLE® IOS®, GOOGLE™ ANDROID™, BLACKBERRY® OS, or the like. The User interface 206 may facilitate display, execution, interaction, manipulation, or operation of program components through textual or graphical facilities. For example, user interfaces may provide computer interaction interface elements on a display system operatively connected to the computer system 300, such as cursors, icons, checkboxes, menus, scrollers, windows, widgets, etc. Graphical User Interfaces (GUIs) may be employed, including, without limitation, Apple® Macintosh® operating systems' Aqua®, IBM® OS/2®, Microsoft® Windows® (e.g., Aero, Metro, etc.), web interface libraries (e.g., ActiveX®, Java®, Javascript®, AJAX, HTML, Adobe® Flash®, etc.), or the like.

[0049] In some embodiments, the computer system 300 may implement the web browser 308 stored program components. The web browser 308 may be a hypertext viewing application, such as MICROSOFT® INTERNET EXPLORER®, GOOGLE™ CHROME™, MOZILLA® FIREFOX®, APPLE® SAFARI®, etc. Secure web browsing may be provided using Secure Hypertext Transport Protocol (HTTPS), Secure Sockets Layer (SSL), Transport Layer Security (TLS), etc. Web browsers 308 may utilize facilities such as AJAX, DHTML, ADOBE® FLASH®, JAVASCRIPT®, JAVA®, Application Programming Interfaces (APIs), etc. In some embodiments, the computer system 300 may implement a mail server stored program component. The mail server may be an Internet mail server such as Microsoft Exchange, or the like. The mail server may utilize facilities such as Active Server Pages (ASP), ACTIVEX®, ANSI® C++/C#, MICROSOFT®, .NET, CGI SCRIPTS, JAVA®, JAVASCRIPT®, PERL®, PHP, PYTHON®, WEBOBJECTS®, etc. The mail server may utilize communication protocols such as Internet Message Access Protocol (IMAP), Messaging Application Programming Interface (MAPI), MICROSOFT® exchange, Post Office Protocol (POP), Simple Mail Transfer Protocol (SMTP), or the like. In some embodiments, the computer system 300 may implement a mail client stored program component. The mail client may be a mail viewing application, such as APPLE® MAIL, MICROSOFT® ENTOURAGE®, MICROSOFT® OUTLOOK®, MOZILLA® THUNDERBIRD®, etc.

[0050] Furthermore, one or more computer-readable storage media may be utilized in implementing embodiments consistent with the present disclosure. A computer-readable storage medium refers to any type of physical memory on which information or data readable by a processor may be stored. Thus, a computer-readable storage medium may store instructions for execution by one or more processors, including instructions for causing the processor(s) to perform steps or stages consistent with the embodiments described herein. The term “computer-readable medium” should be understood to include tangible items and exclude carrier waves and transient signals, i.e., non-transitory. Examples include Random Access Memory (RAM), Read-Only Memory (ROM), volatile memory, non-volatile memory, hard drives, Compact Disc (CD) ROMs, Digital Video Disc (DVDs), flash drives, disks, and any other known physical storage media.

[0051] Finally, the language used in the specification has been principally selected for readability and instructional purposes, and it may not have been selected to delineate or circumscribe the inventive subject matter. Accordingly, the disclosure of the embodiments of the disclosure is intended to be illustrative, but not limiting, of the scope of the disclosure.

[0052] With respect to the use of substantially any plural and/or singular terms herein, those having skill in the art can translate from the plural to the singular and/or from the singular to the plural as is appropriate to the context and/or application. The various singular/plural permutations may be expressly set forth herein for sake of clarity.

[0053] Any of the software components or functions described in this application, may be implemented as software code to be executed by a processor using any suitable computer language such as, for example, Java, C++ or Perl using, for example, conventional or object-oriented techniques. The software code may be stored as a series of instructions, or commands on a computer readable medium, such as a random access memory (RAM), a read only memory (ROM), a magnetic medium such as a hard-drive or a floppy disk, or an optical medium such as a CD-ROM. Any such computer readable medium may reside on or within a single computational apparatus, and may be present on or within different computational apparatuses within a system or network.

[0054] Fig. 4 depicts a flow diagram of an exemplary method 400 for detecting illegitimate money transfers in accordance with the present invention. At step 401, the method includes, updating a scalable graph database with entity information, payment flows, other news that may contain semantic words of interest such as ,but not limited, “fraud” or “crime”, thorough an Extract-transform-Load process (ETL process) with appropriate updates of indexes.

[0055] At step 402, receiving an investigation prompt from an external computer.

[0056] At step 403, checking a payment transaction with past payment transactions in the graph database using various Suspicious Activity Monitoring Rules (SAM rules), or checking the related entity details in the graph database. Pushing the data further into the workflow, at the occurrence of a match.

[0057] At step 404, storing and updating the record in the database, in case a true positive occurs, to be recalled into the workflow if similar transaction or entities are re-triggered in a subsequent transaction.

[0058] The above description is illustrative and is not restrictive. Many variations of the invention may become apparent to those skilled in the art upon review of the disclosure. The scope of the invention can, therefore, be determined not with reference to the above description, but instead can be determined with reference to the pending claims along with their full scope or equivalents.

[0059] One or more features from any embodiment may be combined with one or more features of any other embodiment without departing from the scope of the invention.

[0060] A recitation of "a", "an" or "the" is intended to mean "one or more" unless specifically indicated to the contrary.

[0061] All patents, patent applications, publications, and descriptions mentioned above are herein incorporated by reference in their entirety for all purposes. None is admitted to be prior art.

[0062] Although the invention has been described in detail for the purpose of illustration based on what is currently considered to be the most practical and preferred embodiments, it is to be understood that such detail is solely for that purpose and that the invention is not limited to the disclosed embodiments, but, on the contrary, is intended to cover modifications and equivalent arrangements that are within the spirit and scope of the appended claims. For example, it is to be understood that the present invention contemplates that, to the extent possible, one or more features of any embodiment can be combined with one or more features of any other embodiment.

ABSTRACT

USE OF GRAPH DATABASES TO DETECT ILLEGITIMATE MONEY TRANSFERS

The present disclosure includes a method that comprises of determining an entity identifier and a geographic location identifier. The method further includes forming an article identifier using the entity identifier and the geographic location identifier. The articles pertaining to the entity identifier and the geographic location identifier can be grouped. The articles can then be stored in an article database that is indexed to the entity identifier and the geographic location identifier. A suspicious entity in a graph database can be identified. Entities associated with the suspicious entity can be identified using the graph database. The method can then include retrieving articles from the article database associated with the suspicious entity and the determined entities to determine if the suspicious entity is engaged in fraudulent activity.

1/4

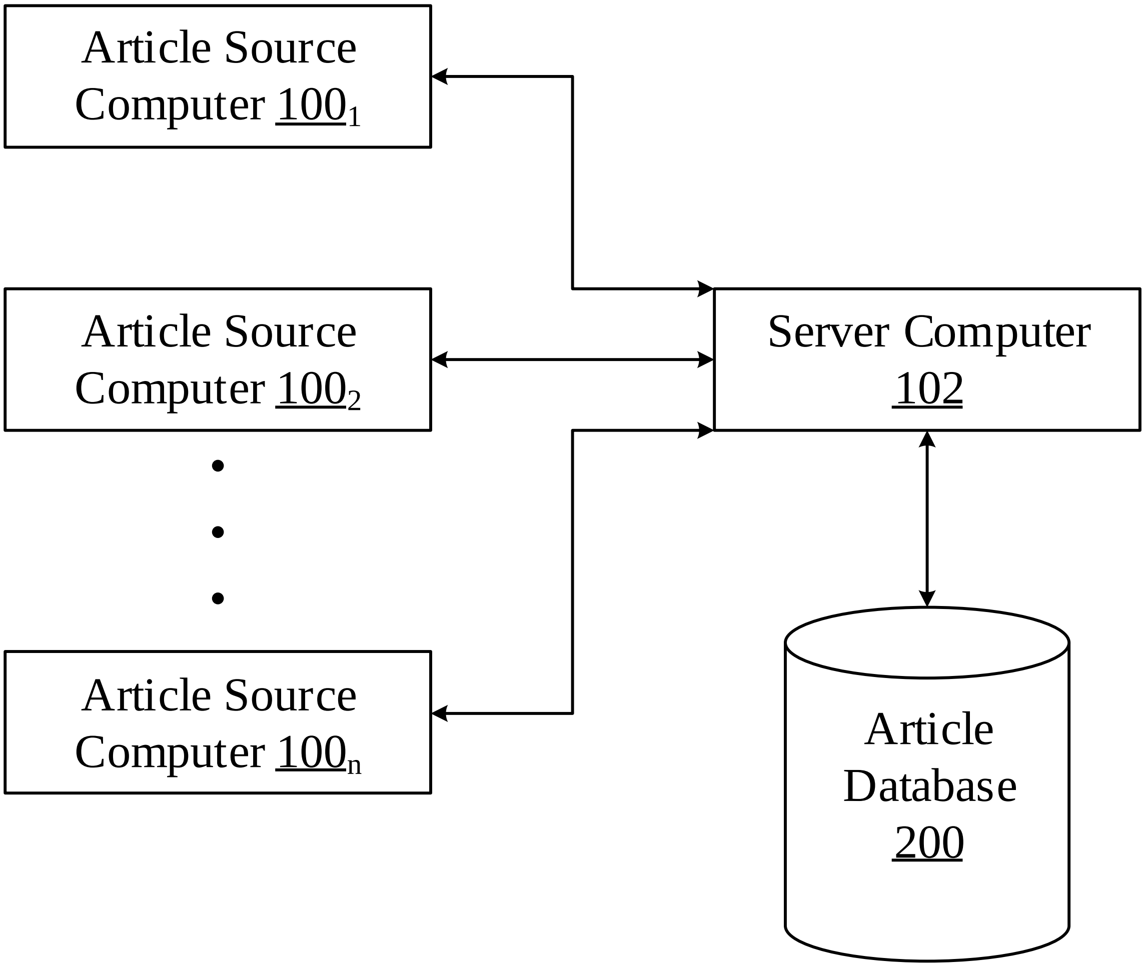
Article Source Computer $\underline{100}_1$
Article Source Computer $\underline{100}_2$
Article Source Computer $\underline{100}_n$
Server Computer $\underline{102}$
Article Database $\underline{200}$

Fig. 1

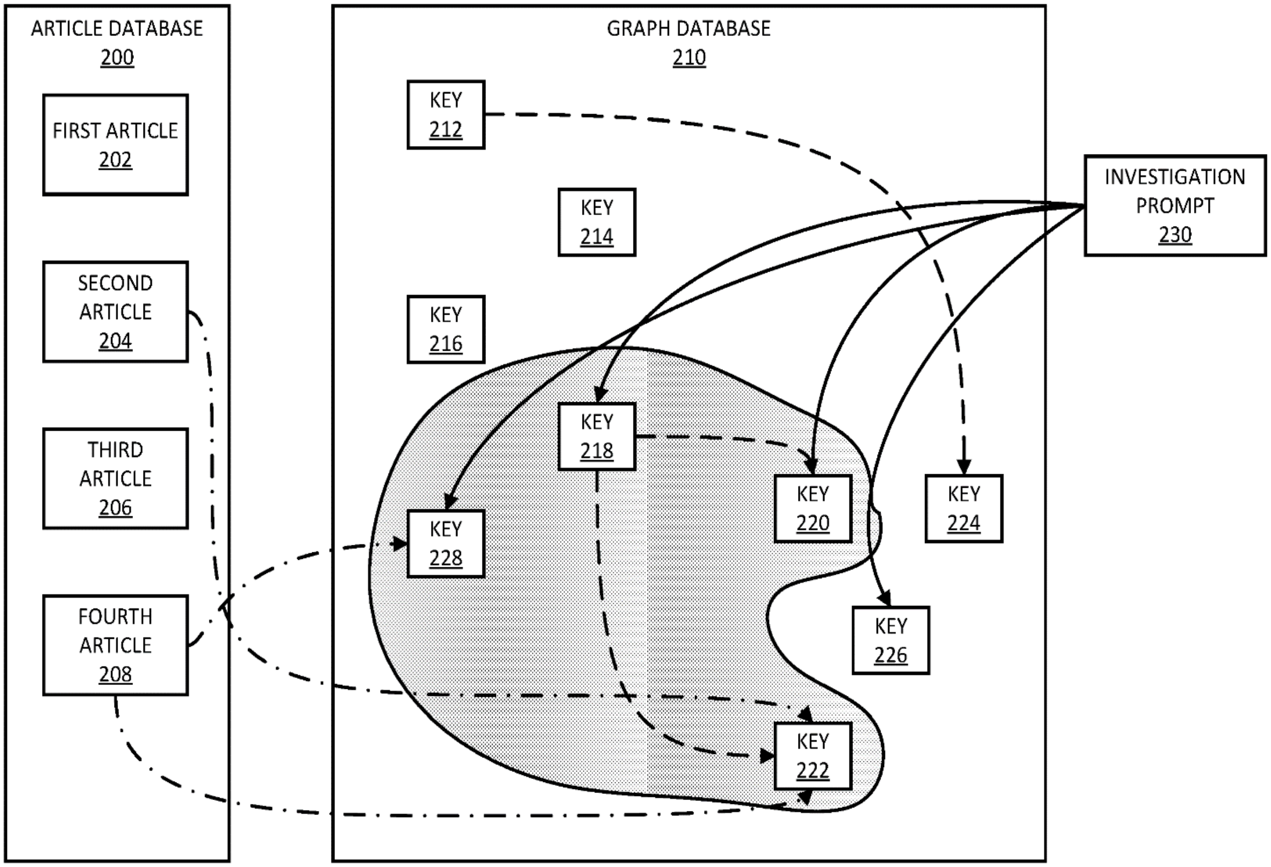
ARTICLE DATABASE
200
FIRST ARTICLE
202
SECOND ARTICLE
204
THIRD ARTICLE
206
FOURTH ARTICLE
208
GRAPH DATABASE
210
KEY
212
KEY
214
KEY
216
KEY
218
KEY
220
KEY
224
KEY
228
KEY
226
KEY
222
INVESTIGATION PROMPT
230

FIG. 2

3/4

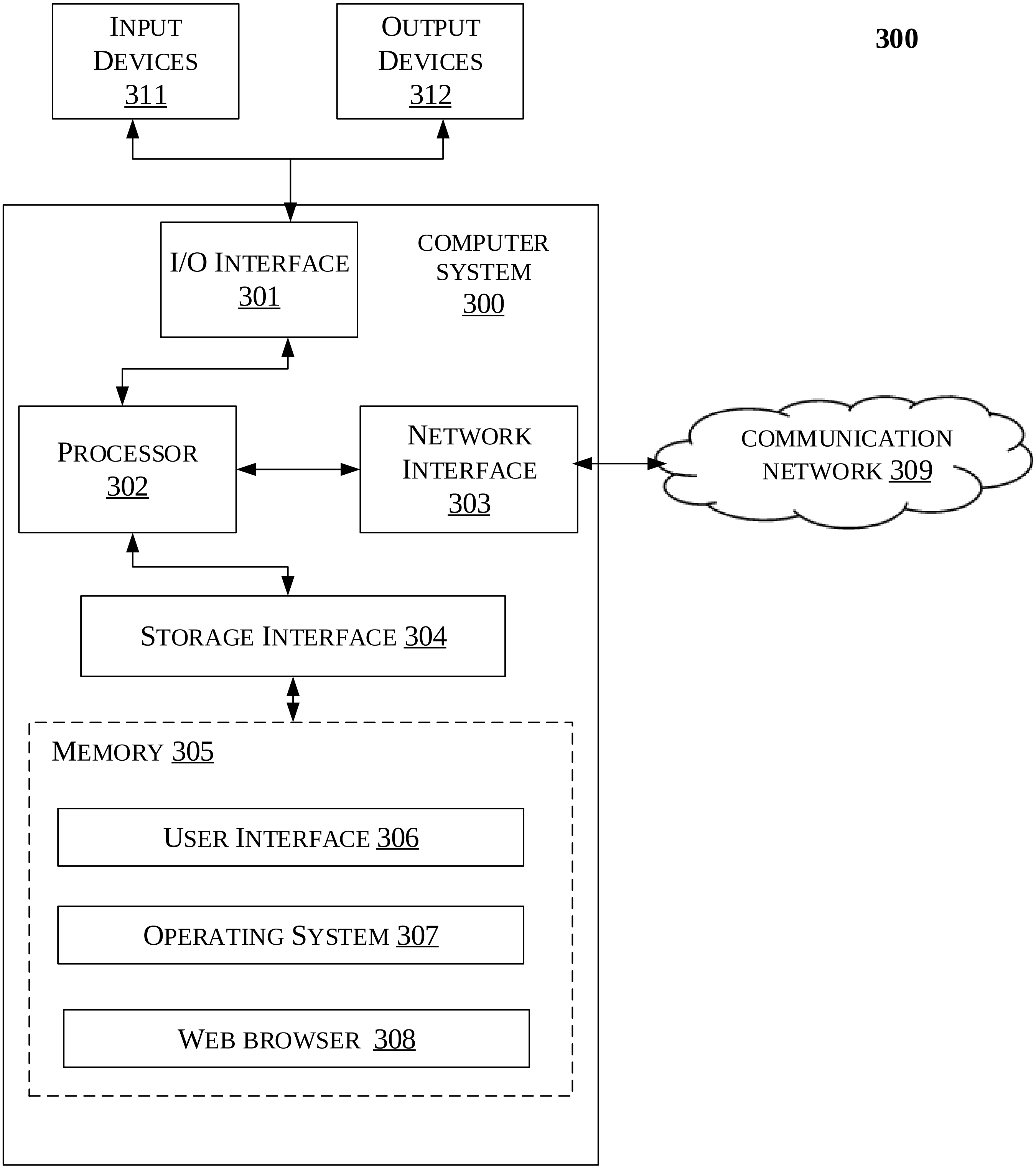
300
INPUT DEVICES 311
OUTPUT DEVICES 312
I/O INTERFACE 301
COMPUTER SYSTEM 300
PROCESSOR 302
NETWORK INTERFACE 303
COMMUNICATION NETWORK 309
STORAGE INTERFACE 304
MEMORY 305
USER INTERFACE 306
OPERATING SYSTEM 307
WEB BROWSER 308

Fig. 3

4/4

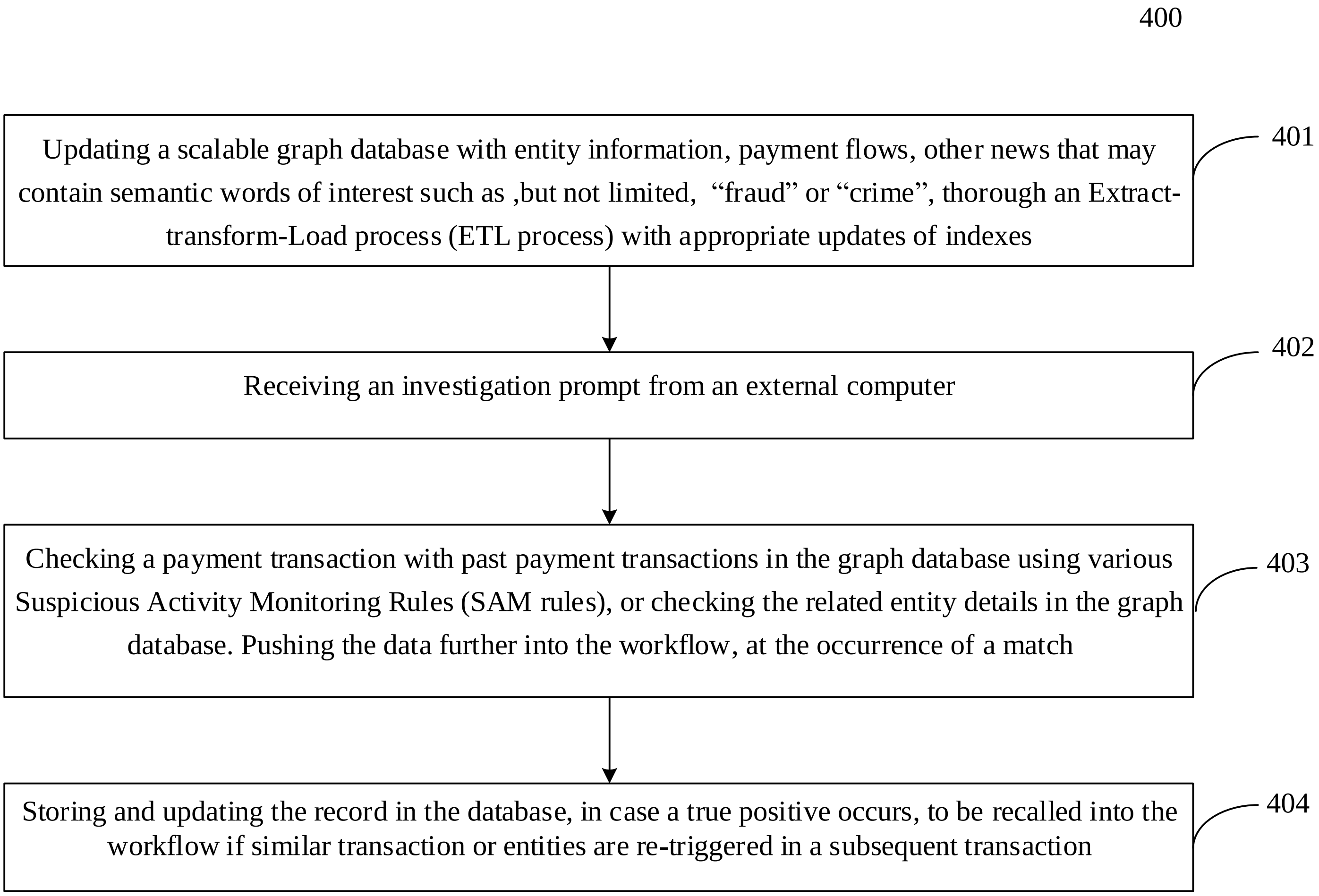
400
401
Updating a scalable graph database with entity information, payment flows, other news that may contain semantic words of interest such as ,but not limited, “fraud” or “crime”, thorough an Extract-transform-Load process (ETL process) with appropriate updates of indexes
402
Receiving an investigation prompt from an external computer
403
Checking a payment transaction with past payment transactions in the graph database using various Suspicious Activity Monitoring Rules (SAM rules), or checking the related entity details in the graph database. Pushing the data further into the workflow, at the occurrence of a match
404
Storing and updating the record in the database, in case a true positive occurs, to be recalled into the workflow if similar transaction or entities are re-triggered in a subsequent transaction

Fig. 4