

AUTHENTICATION METHOD AND SYSTEM

FIELD OF INVENTION

[001] The present invention relates broadly, but not exclusively, to authentication methods and systems.

BACKGROUND

[002] electronic-Know Your Customer (eKYC) is a digital due diligence process performed by a business entity or service provider to verify the identity of its clients and to assess potential risks of illegal intentions towards the business relationship (e.g. money laundering).

[003] Fraudsters are a serious problem for eKYC processes. In many cases, fraudsters come as a group to maximize the benefits, but sometimes it is difficult to detect all fraudsters from the same group.

[004] Currently, there are a number of methods to detect fraudsters. Some solutions only consider relationships that are already known between identities, and completely ignore potential links. However, the known relationships are a small portion compared to the overall links. Therefore, if fraudster detection is based on known relationships only, the detection result can only cover a small portion. Some other solutions only consider direct links with fraudsters, which also cover a small portion and do not make use of the complex relationship among identities.

SUMMARY

[005] Embodiments seek to provide authentication methods and systems by finding a relationship between a new identity and a known fraudster. In an embodiment, an authentication method involves graph link prediction and graph label propagation techniques to find unknown relationships among identities. A graph is built based on historical data, wherein every identity is treated as a node, and relationships are used as edges. In this way, a link prediction model can be trained with the historical data. Thereafter, the links between every two nodes can be predicted to find unknown relationship(s). Subsequently, based on the new graph (i.e. with the predicted links), label propagation techniques can be used to propagate the fraudster labels from those

that are already known, to those that were initially unknown but can be predicted by the links.

[006] According to another embodiment, there is provided an authentication method, including the following steps: constructing a graph to represent a set of data, the set of data comprising identities and known relationships between the identities, wherein each node in the graph corresponds to each of the identities and each edge in the graph corresponds to each of the known relationships; predicting unknown relationships between the identities using a model that is trained based on a link prediction algorithm; updating the graph to further comprise at least one new edge that corresponds to a predicted relationship, wherein the updated graph comprises at least one node that corresponds to an identity with a fraudster label; and labeling at least one other identity with the fraudster label based on a label propagation algorithm.

[007] According to another embodiment, there is provided an authentication system, including: a processor module; and a memory module including computer program code; the memory module and the computer program code configured to, with the processor module, cause the system at least to: construct a graph to represent a set of data, the set of data comprising identities and known relationships between the identities, wherein each node in the graph corresponds to each of the identities and each edge in the graph corresponds to each of the known relationships; predict unknown relationships between the identities using a model that is trained based on a link prediction algorithm; update the graph to further comprise at least one new edge that corresponds to a predicted relationship, wherein the updated graph comprises at least one node that corresponds to an identity with a fraudster label; and label at least one other identity with the fraudster label based on a label propagation algorithm.

BRIEF DESCRIPTION OF THE DRAWINGS

[008] Embodiments are provided by way of example only, and will be better understood and readily apparent to one of ordinary skill in the art from the following written description, read in conjunction with the drawings, in which:

[009] Figure 1 is a flow chart illustrating a authentication method, according to an embodiment.

[0010] Figure 2 is a constructed graph according to an embodiment.

[0011] Figure 3 is a constructed graph with a predicted link according to an embodiment.

[0012] Figure 4 is a constructed graph with a predicted link and newly labeled fraudsters according to an embodiment.

[0013] Figure 5 is a constructed graph with a predicted link and an additional newly labeled fraudster according to an embodiment.

[0014] Figure 6 is a schematic of an authentication system according to an embodiment.

[0015] Figure 7 shows a schematic diagram of a computer system suitable for use in executing at least some steps of the authentication method.

DETAILED DESCRIPTION

[0016] Embodiments will be described, by way of example only, with reference to the drawings. Like reference numerals and characters in the drawings refer to like elements or equivalents.

[0017] Some portions of the description which follows are explicitly or implicitly presented in terms of algorithms and functional or symbolic representations of operations on data within a computer memory. These algorithmic descriptions and functional or symbolic representations are the means used by those skilled in the data processing arts to convey most effectively the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of steps leading to a desired result. The steps are those requiring physical manipulations of physical quantities, such as electrical, magnetic or optical signals capable of being stored, transferred, combined, compared, and otherwise manipulated.

[0018] Unless specifically stated otherwise, and as apparent from the following, it will be appreciated that throughout the present specification, discussions utilizing terms such as “receiving”, “scanning”, “calculating”, “determining”, “replacing”, “generating”, “initializing”, “outputting”, or the like, refer to the action and processes of a computer system, or similar electronic device, that manipulates and transforms data represented as physical quantities within the computer system into other data similarly represented

as physical quantities within the computer system or other information storage, transmission or display devices.

[0019] The present specification also discloses apparatus for performing the operations of the methods. Such apparatus may be specially constructed for the required purposes, or may comprise a computer or other device selectively activated or reconfigured by a computer program stored in the computer. The algorithms and displays presented herein are not inherently related to any particular computer or other apparatus. Various machines may be used with programs in accordance with the teachings herein. Alternatively, the construction of more specialized apparatus to perform the required method steps may be appropriate. The structure of a computer suitable for executing the various methods / processes described herein will appear from the description below.

[0020] In addition, the present specification also implicitly discloses a computer program, in that it would be apparent to the person skilled in the art that the individual steps of the method described herein may be put into effect by computer code. The computer program is not intended to be limited to any particular programming language and implementation thereof. It will be appreciated that a variety of programming languages and coding thereof may be used to implement the teachings of the disclosure contained herein. Moreover, the computer program is not intended to be limited to any particular control flow. There are many other variants of the computer program, which can use different control flows without departing from the spirit or scope of the invention.

[0021] Furthermore, one or more of the steps of the computer program may be performed in parallel rather than sequentially. Such a computer program may be stored on any computer readable medium. The computer readable medium may include storage devices such as magnetic or optical disks, memory chips, or other storage devices suitable for interfacing with a computer. The computer readable medium may also include a hard-wired medium such as exemplified in the Internet system, or wireless medium such as exemplified in the GSM mobile telephone system. The computer program when loaded and executed on such a computer effectively results in an apparatus that implements the steps of the preferred method.

[0022] Authentication can be considered as a form of fraud detection in which legitimacy of users are verified and potential fraudsters may be detected before fraudulent acts are carried out. Effective authentication can enhance data security of systems so that digital data is protected from unauthorized users. Embodiments seek to provide authentication methods and systems by finding a relationship between a new identity and a known fraudster. In an embodiment, an authentication method involves graph link prediction and graph label propagation techniques to find unknown relationships among identities and consequently to find fraudsters. Authentication can be implemented remotely using a remote authentication server, as embodied by authentication system 600 that is described in detailed below. Remote authentication allows fraud detection to be performed centrally (typically requiring less resources) and away from user terminals over unsecure communication channels.

[0023] Graph link prediction is a technique to predict future possible links in a network or to predict missing links due to incomplete data. Given a constructed graph, link prediction predicts the probability of a link between two nodes. The link prediction task is divided into two categories. The first category is to predict whether the new link appears in the future. The second category is to predict a hidden unknown link.

[0024] Given nodes and their known links (neighbors), it is possible to train a link prediction model to generate a link potential score for two given nodes. If the score is above a certain threshold, the two nodes can be considered to be linked. There are different methods to train the model.

[0025] For example, in one method, the input of the model is the whole graph, with its nodes and links. The model can be (i) a graph neural network (GNN) like GATNE or (ii) a graph convolutional network (GCN). The model is able to model the nodes' information based on its neighbors and train the model by link prediction. After training, every node can get an embedding by the model. In this manner, it is possible to use a similarity score (cosine score) to decide whether there is a link between two nodes if the similarity score is above a threshold.

[0026] As another example, the input of the model is a node pair, e.g. (Info_Node_A, and Info_Node_B), which contains the information of the node itself and its neighbors. Then, it is possible to train a classifier model to predict if a link exists or not. After training, the model is fed with two nodes' information to get a result. The result can be a score (above a threshold is "yes", otherwise no), or a certain "yes"/"no".

[0027] Graph label propagation is technique that uses a semi-supervised machine learning algorithm that assign labels to previously unlabeled data points to propagate labels from known points. At the start of the algorithm, a (generally small) subset of the data points have labels. These labels are propagated to the unlabeled data points throughout the course of the algorithm. At initial condition, the nodes carry a label that denotes the community they belong to. Membership in a community changes based on the labels that the neighboring nodes possess. This change is subject to the maximum number of labels within one degree of the nodes. Every node is initialized with a unique label, then the labels diffuse/propagate through the network.

[0028] The objective of graph label propagation is to update a current node by its previous status and its neighbors. With (a) known fraudsters nodes, and (b) known and predicted links, at every time step, it is possible to update the score of a certain node based on the score of itself and the scores of its neighbors. In one implementation, a graph label propagation formula is: *Current status = function(Previous status, statuses of neighbors)*. As the update runs at every time step, the status of neighbors can be a neighbors' current status or previous status. The steps can be re-executed several times, until a maximum step number, or when the scores of the nodes do not change anymore. An example is updated score = max(previous score, scores of neighbors), which directly propagates the fraudster label among the graph. The function can be more complex, for example, updated score = $0.6 * \text{previous score} + 0.4 * \text{average of scores of neighbors}$. An even more complex function can be a multilayer perceptron (MLP) network that can be trained.

[0029] A graph is built based on historical data, wherein every identity is treated as a node, and relationships are used as edges. Once the graph is built, a link prediction algorithm is chosen and a model can be trained using training data. Thereafter, the links between every two nodes can be predicted to find unknown relationship(s). Subsequently, based on the new graph (i.e. with the predicted links), label propagation techniques can be used to propagate the fraudster labels from those that are already known, to those that were initially unknown but can be predicted by the links.

[0030] By using graph link prediction techniques to find potential relationships between identities that were initially unknown, it is possible to cover more identities and consequently detect more fraudsters. Furthermore, by using label propagation

techniques to propagate fraudster labels from those that are already known to those that were initially unknown, it is possible to further improve the fraudster detection scope.

[0031] The techniques described in this specification produce one or more technical effects. First, by converting identities and relationships between identities into a graph, it is possible to implement graph techniques to improve detection performance. Second, by using graph link prediction techniques to find potential relationships between identities that were initially unknown, it is possible to cover more identities that did not have any links known beforehand. Third, by using label propagation techniques to propagate fraudster labels from those already known to those that were initially unknown (based on the links that are known and predicted), it is possible to further improve the fraudster detection scope.

[0032] Figure 1 is a flow chart 100 illustrating an authentication method, according to an embodiment. At step 102, a graph is constructed to represent a set of data. The set of data can be historical data, and the set of data comprises identities and known relationships between the identities. Each node in the graph corresponds to each of the identities while each edge in the graph corresponds to each of the known relationships.

[0033] Figure 2 is a constructed graph according to an embodiment. The set of data comprises four identities (identity 1, identity 2, identity 3 and identity 4) and known relationships between the identities. Each node in the graph corresponds to each of the identities while each edge in the graph corresponds to each of the known relationships. The relationship/link between identity 1 and identity 2 is a similar face and same ID number (e.g. passport number, social security number, national registration ID card number, etc.). This relationship/link between identity 1 and identity 2 is represented by one edge in the graph. Similarly, the relationship/link between identity 2 and identity 3 is a same device (e.g. same mobile phone model). This relationship/link between identity 2 and identity 3 is represented by another edge in the graph. Likewise, the relationship/link between identity 2 and identity 4 is a same device. This relationship/link between identity 2 and identity 4 is represented by yet another edge in the graph. The relationship/link between identity 3 and identity 4 is a same device (e.g. same mobile phone model) and same ID number. This relationship/link between identity 3 and identity 4 is represented by another edge in the graph. In Figure 2, identity 4 is a known fraudster and is labelled accordingly. Currently, it is not known whether identity 1, identity 2 and identity 3 are fraudsters.

[0034] The known relationships may be based on similarity of a characteristic of the identities. For example, the characteristic of the identities can be: biometric data (such as facial data), identity (ID) document data and/or personal electronic device data (e.g. mobile phone model).

[0035] Referring back to Figure 1, at step 104, unknown relationships between the identities are predicted using a neural network model that is trained based on a link prediction algorithm.

[0036] Optionally, before step 104, step 103 involves training the model with a set of data of training data. The training data can be the above-mentioned historical data. The link prediction model learns the information and logic inside every link between the nodes to provide the predictive result. Accordingly, for every two nodes that have no link/relationship, the link prediction model can predict whether or not there is a potential link. The predicted links (i.e. new edges) are added to the graph.

[0037] Referring back to Figure 1, at step 106, the graph is updated to further comprise at least one new edge that corresponds to a predicted relationship. Figure 3 is a constructed graph with a predicted link (indicated by dashed lines) according to an embodiment. Initially, there is no relationship between identity 1 and 3 as shown in Figure 2. The trained model is able to predict that there is a relationship between identity 1 and 3 (indicated by the dashed lines in Figure 3).

[0038] The updated graph comprises at least one node that corresponds to an identity with a fraudster label. With reference to Figure 3, identity 4 is a known fraudster and is labelled accordingly.

[0039] At step 108, at least one other identity is labeled with the fraudster label based on a label propagation algorithm. In other words, based on the updated graph, the label propagation algorithm is used to propagate the fraudster label from those that are known to those that are unknown. Figure 4 is a constructed graph with a predicted link and newly labeled fraudsters according to an embodiment. With reference to Figure 4, the label propagation algorithm propagates the fraudster label from identity 4 to identity 2 and identity 3. In other words, initially it was not known whether identity 2 and identity 3 were fraudsters but after the label propagation algorithm is executed, identity 2 and identity 3 are labelled accordingly. The label propagation algorithm propagates the

fraudster label from those that are known (i.e. identity 4) to those that are not known (identity 2 and identity 3) but can be predicted based on the links.

[0040] Figure 5 is a constructed graph with a predicted link and an additional newly labeled fraudster according to an embodiment. With reference to Figure 5, the label propagation algorithm further propagates the fraudster label from identity 2, identity 3 and identity 4 to identity 1. In other words, initially it was not known whether identity 1 was a fraudster but after further execution of the label propagation algorithm, identity 1 is labelled accordingly. The label propagation algorithm propagates the fraudster label from those that are known (i.e. identity 2, identity 3, and identity 4) to those that are not known (identity 1) but can be predicted based on the links. It is to be noted that the predicted link (predicted at step 104) helps to propagate the fraudster label to identity 1 and detect identity 1 as a fraudster. In the authentication method, authentication of the at least one other identity with the fraudster label may be refused. In this manner, data security and system integrity can be enhanced as unauthorized users are prevented from accessing the system.

[0041] Accordingly, comparing the initial state shown in Figure 2 to the final state shown in Figure 5, identity 1, identity 2 and identity 3 have been labelled as fraudsters. Accordingly, fraudster detection is accomplished as identity 1, identity 2 and identity 3 have been recognized as fraudsters.

[0042] Optionally, at step 110, the predicted relationship (e.g. between identity 1 and identity 3 as shown in Figure 5) and the at least one fraudster label (e.g. the newly labeled fraudsters identity 1, identity 2 and identity 3) are added to the set of data to form a revised set of data so that the model can be re-trained based on the link prediction algorithm with the revised set of data. In this manner, it is possible to enhance the training process.

[0043] Figure 6 shows an authentication system according to an embodiment. The authentication system 600 includes a processor module 602 and a memory module 604 including computer program code. The memory module 604 and the computer program code are configured to, with the processor module 602, cause the system 600 at least to: (i) construct a graph to represent a set of data, the set of data including identities and known relationships between the identities, wherein each node in the graph corresponds to each of the identities and each edge in the graph corresponds to each of the known relationships; (ii) predict unknown relationships between the

identities using a model that is trained based on a link prediction algorithm; (iii) update the graph to further comprise at least one new edge that corresponds to a predicted relationship, wherein the updated graph comprises at least one node that corresponds to an identity with a fraudster label; and (iv) label at least one other identity with the fraudster label based on a label propagation algorithm. The system may be further caused to refuse authentication of the at least one other identity with the fraudster label. The aforementioned model is trained based on the link prediction algorithm with the set of data.

[0044] The predicted relationship and the at least one fraudster label can be added to the set of data to form a revised set of data. The system 600 can be further caused to re-train the model based on the link prediction algorithm with the revised set of data.

[0045] The aforementioned known relationships may be based on similarity of a characteristic of the identities, e.g. biometric data, identity document data and/or personal electronic device data.

[0046] Figure 7 shows a schematic diagram of a computer system suitable for use in executing at least some steps of the authentication method.

[0047] The following description of the computer system / computing device 700 is provided by way of example only and is not intended to be limiting.

[0048] As shown in Figure 7, the example computing device 700 includes a processor 704 for executing software routines. Although a single processor is shown for the sake of clarity, the computing device 700 may also include a multi-processor system. The processor 704 is connected to a communication infrastructure 706 for communication with other components of the computing device 700. The communication infrastructure 706 may include, for example, a communications bus, cross-bar, or network.

[0049] The computing device 700 further includes a main memory 708, such as a random access memory (RAM), and a secondary memory 710. The secondary memory 710 may include, for example, a hard disk drive 712 and/or a removable storage drive 714, which may include a magnetic tape drive, an optical disk drive, or the like. The removable storage drive 714 reads from and/or writes to a removable storage unit 718 in a well-known manner. The removable storage unit 718 may include a magnetic tape, optical disk, or the like, which is read by and written to by removable

storage drive 714. As will be appreciated by persons skilled in the relevant art(s), the removable storage unit 718 includes a computer readable storage medium having stored therein computer executable program code instructions and/or data.

[0050] In an alternative embodiment, the secondary memory 710 may additionally or alternatively include other similar devices for allowing computer programs or other instructions to be loaded into the computing device 700. Such devices can include, for example, a removable storage unit 722 and an interface 720. Examples of a removable storage unit 722 and interface 720 include a removable memory chip (such as an EPROM or PROM) and associated socket, and other removable storage units 722 and interfaces 720 which allow software and data to be transferred from the removable storage unit 722 to the computer system 700.

[0051] The computing device 700 also includes at least one communication interface 724. The communication interface 724 allows software and data to be transferred between computing device 700 and external devices via a communication path 726. In various embodiments, the communication interface 724 permits data to be transferred between the computing device 700 and a data communication network, such as a public data or private data communication network. The communication interface 724 may be used to exchange data between different computing devices 700 which such computing devices 700 form part an interconnected computer network. Examples of a communication interface 724 can include a modem, a network interface (such as an Ethernet card), a communication port, an antenna with associated circuitry and the like. The communication interface 724 may be wired or may be wireless. Software and data transferred via the communication interface 724 are in the form of signals which can be electronic, electromagnetic, optical or other signals capable of being received by communication interface 724. These signals are provided to the communication interface via the communication path 726.

[0052] Optionally, the computing device 700 further includes a display interface 702 which performs operations for rendering images to an associated display 730 and an audio interface 732 for performing operations for playing audio content via associated speaker(s) 734.

[0053] As used herein, the term "computer program product" may refer, in part, to removable storage unit 718, removable storage unit 722, a hard disk installed in hard disk drive 712, or a carrier wave carrying software over communication path 726

(wireless link or cable) to communication interface 724. Computer readable storage media refers to any non-transitory tangible storage medium that provides recorded instructions and/or data to the computing device 700 for execution and/or processing. Examples of such storage media include floppy disks, magnetic tape, CD-ROM, DVD, Blu-ray™ Disc, a hard disk drive, a ROM or integrated circuit, USB memory, a magneto-optical disk, or a computer readable card such as a PCMCIA card and the like, whether or not such devices are internal or external of the computing device 700. Examples of transitory or non-tangible computer readable transmission media that may also participate in the provision of software, application programs, instructions and/or data to the computing device 700 include radio or infra-red transmission channels as well as a network connection to another computer or networked device, and the Internet or Intranets including e-mail transmissions and information recorded on Websites and the like.

[0054] The computer programs (also called computer program code) are stored in main memory 708 and/or secondary memory 710. Computer programs can also be received via the communication interface 724. Such computer programs, when executed, enable the computing device 700 to perform one or more features of embodiments discussed herein. In various embodiments, the computer programs, when executed, enable the processor 704 to perform features of the above-described embodiments. Accordingly, such computer programs represent controllers of the computer system 700.

[0055] Software may be stored in a computer program product and loaded into the computing device 700 using the removable storage drive 714, the hard disk drive 712, or the interface 720. Alternatively, the computer program product may be downloaded to the computer system 700 over the communications path 726. The software, when executed by the processor 704, causes the computing device 700 to perform functions of embodiments described herein.

[0056] It is to be understood that the embodiment of Figure 7 is presented merely by way of example. Therefore, in some embodiments one or more features of the computing device 700 may be omitted. Also, in some embodiments, one or more features of the computing device 700 may be combined together. Additionally, in some embodiments, one or more features of the computing device 700 may be split into one or more component parts.

[0057] This specification uses the term “configured to” in connection with systems, devices, and computer program components. For a system of one or more computers to be configured to perform particular operations or actions means that the system has installed on it software, firmware, hardware, or a combination of them that in operation cause the system to perform the operations or actions. For one or more computer programs to be configured to perform particular operations or actions means that the one or more programs include instructions that, when executed by data processing apparatus, cause the apparatus to perform the operations or actions. For special-purpose logic circuitry to be configured to perform particular operations or actions means that the circuitry has electronic logic that performs the operations or actions.

[0058] It will be appreciated by a person skilled in the art that numerous variations and/or modifications may be made to the present invention as shown in the specific embodiments without departing from the spirit or scope of the invention as broadly described. The present embodiments are, therefore, to be considered in all respects to be illustrative and not restrictive.

CLAIMS

1. An authentication method, comprising:
 - constructing a graph to represent a set of data, the set of data comprising identities and known relationships between the identities, wherein each node in the graph corresponds to each of the identities and each edge in the graph corresponds to each of the known relationships;
 - predicting unknown relationships between the identities using a model that is trained based on a link prediction algorithm;
 - updating the graph to further comprise at least one new edge that corresponds to a predicted relationship, wherein the updated graph comprises at least one node that corresponds to an identity with a fraudster label; and
 - labeling at least one other identity with the fraudster label based on a label propagation algorithm.
2. The method as claimed in claim 1, wherein the model is trained based on the link prediction algorithm with the set of data.
3. The method as claimed in claim 2, wherein the predicted relationship and the at least one fraudster label are added to the set of data to form a revised set of data, and wherein the method further comprises re-training the model based on the link prediction algorithm with the revised set of data.
4. The method as claimed in any one of the preceding claims, wherein the known relationships are based on similarity of a characteristic of the identities.
5. The method as claimed in claim 4, wherein the characteristic of the identities comprises: biometric data, identity document data and/or personal electronic device data.
6. The method as claimed in any one of the preceding claims, further comprising refusing authentication of the at least one other identity with the fraudster label.
7. An authentication system, comprising:
 - a processor module; and
 - a memory module including computer program code;

the memory module and the computer program code configured to, with the processor module, cause the system at least to:

construct a graph to represent a set of data, the set of data comprising identities and known relationships between the identities, wherein each node in the graph corresponds to each of the identities and each edge in the graph corresponds to each of the known relationships;

predict unknown relationships between the identities using a model that is trained based on a link prediction algorithm;

update the graph to further comprise at least one new edge that corresponds to a predicted relationship, wherein the updated graph comprises at least one node that corresponds to an identity with a fraudster label; and

label at least one other identity with the fraudster label based on a label propagation algorithm.

8. The system as claimed in claim 7, wherein the model is trained based on the link prediction algorithm with the set of data.

9. The system as claimed in claim 8, wherein the predicted relationship and the at least one fraudster label are added to the set of data to form a revised set of data, and wherein the system is further caused to re-train the model based on the link prediction algorithm with the revised set of data.

10. The system as claimed in any one of claims 7 to 9, wherein the known relationships are based on similarity of a characteristic of the identities.

11. The system as claimed in claim 10, wherein the characteristic of the identities comprises: biometric data, identity document data and/or personal electronic device data.

12. The system as claimed in any one of claims 7 to 11, wherein the system is further caused to refuse authentication of the at least one other identity with the fraudster label.