

## **HYBRID QUANTUM-CLASSICAL AUTOENCODER FOR CYBERSECURITY ANOMALY DETECTION**

### **TECHNICAL FIELD**

[0001] The present disclosure relates generally to cybersecurity, and more specifically, to methods of anomaly detection using a hybrid quantum-classical autoencoder.

### **BACKGROUND**

[0002] A cyberattack can be defined as any offensive action that targets computer information systems, computer networks, infrastructures, personal computer devices or smartphones. An attacker is a person or process that attempts to access data, functions, or other restricted areas of the system without authorization, often with malicious intent. A cyberattack can be employed by sovereign states, individuals, groups, societies or organizations and it may originate from an anonymous source. Cyberattacks have increased over recent years in big part because of the increase in use of Internet of Things (IoT) devices.

[0003] Electronic information across networks is a crucial aspect of enterprise or e-commerce systems. However, malicious or unauthorized use of these systems is on the rise, as evidenced by reports of breach and fraud, despite implementation of existing security systems. During the first half of 2021, attacks on IoT devices more than doubled. A well-known example of a cyberattack is a denial-of-service attack (DoS attack). In a DoS attack, a perpetrator makes a machine or network resource unavailable to its users by disrupting services of a host connected to a network. Denial of service is typically accomplished by flooding the targeted machine or resource with superfluous requests in an attempt to overload systems and prevent some or all legitimate requests from being fulfilled. Similarly, in a distributed denial-of-service attack (DDoS attack), the incoming traffic flooding the victim originates from many different sources. More sophisticated strategies are required to mitigate this type of attack; simply attempting to block a single source is insufficient as there are multiple sources.

[0004] In addition to disrupting service, a cyberattack can steal, alter, or destroy a specified target by hacking into a private network or otherwise susceptible system. Cyberattacks can range from installing spyware on a personal computer to attempting to destroy the infrastructure.

[0005] When not properly secured, IoT devices can introduce a number of new and preventable attack vectors. Some of these risks are relatively simple, such as unpatched operating systems or insecure passwords that make easy targets. As organizations in virtually every industry increase their operational reliance on IoT devices, security teams must account for both simple and complex risk factors specific to the world of IoT.

[0006] Advanced persistent threats (APTs) which may target the exfiltration of critical data, typically include a series of steps: infection, exploitation, command and control, lateral movement and data exfiltration. In the “command and control” phase, an attacker maintains a communication channel between an infected host inside the targeted organization and a remote server controlled by the attacker. This phase can span weeks or months. However, despite its long duration, its detection in real-world organizations remains a great challenge. In fact, to further frustrate detection efforts, some attackers may not only minimize their footprint by combining active with stealthy phases, but also establish communication channels via unblocked services and protocols, therefore blending in with legitimate traffic. Because most organizations allow their employees to freely browse the Internet, web traffic is a very effective channel for attackers to communicate and maintain control over infected machines.

[0007] Descriptive studies show that, when analyzed over a period of several weeks, web-based command and control traffic patterns exhibit distinctive network profiles, with the frequency and network profile being dependent on the specific threat, or malware family involved in the attack. For example, infected machines may periodically attempt to communicate with the remote server(s) and may generally establish lightweight connections in which they receive new instructions. In a small fraction of these

connections, the infected machine will download a larger amount of data, corresponding to a software update.

[0008] Efforts to detect and prevent cyberattacks have improved. However, at the same time, instigators of cyberattacks have become more sophisticated in their tactics. Recent attempts to improve cybersecurity include the use of machine learning and artificial intelligence (AI).

[0009] Although promising, methods of using machine learning and AI have limitations. For example, most machine learning-based attempts to detect command and control focus on the analysis of individual connections. Given the large volume of data generated today at most organizations' perimeters and the number of entities that must be monitored and analyzed, it is difficult to train models with behavioral patterns observed over weeks (or months) of data. Depending on the size and activity of an entity, perimeter devices such as next generation firewalls can generate more than one terabyte (TB) of log data and involve tens of millions of entries per day. As such, there is a need for improved cyber security systems, and particular for security systems capable of handling large volumes of data and detecting threat patterns exhibited over extended periods of time.

[0010] A "classical processor" uses bits to operate various programs. Their power increases linearly as more bits are added. In contrast, a quantum computer uses qubits to run multidimensional quantum algorithms. Their processing power increases exponentially as qubits are added. Because of this difference, classical computers have much less computing power than quantum computers. Interest in quantum computing has recently surged, in part due to a wave of advances in quantum-computing performance.

[0011] By using quantum computing, the present invention provides improvements over conventional efforts of detecting and preventing cyberattacks. Embodiments include a hybrid quantum-classical autoencoder for cybersecurity anomaly detection.

### SUMMARY OF INVENTION

[0012] The following summary is provided to facilitate an understanding of some of the innovative features unique to the disclosed embodiment and is not intended to be a full description. A full appreciation of the various aspects of the embodiments disclosed herein can be gained by taking into consideration the entire specification, claims, drawings, and abstract as a whole.

[0013] Disclosed herein are systems and methods for cybersecurity that use a hybrid quantum-classical autoencoder.

[0014] Embodiments include a hybrid quantum-classical autoencoder that includes (a) a classical computing component and (b) a quantum computing component. In aspects, the hybrid quantum-classical autoencoder is configured to combine encoding layers of a variational autoencoder coupled with a quantum circuit as a bottleneck with a decoder for training. In aspects, the hybrid quantum-classical computer is configured to detect an anomaly from a data source.

[0015] Embodiments also include a system for detecting anomalies for cybersecurity. The system can include a first autoencoder that uses classical computing, a quantum circuit and a decoder that uses quantum computing. In aspects, the system is configured to combine encoding layers of autoencoder coupled with the quantum circuit as a bottleneck with a decoder for training. In aspects, the system is configured to detect anomalies from a data source (e.g., network data).

[0016] In aspects, the system is constructed by (a) providing an input for an input layer by concatenation of feature values from a sample, (b) training the first autoencoder to obtain trained weights corresponding to cybersecurity data, (c) constructing the encoder with the trained weights layers, (d) constructing and configuring the quantum circuit, (e) adding the quantum circuit to the encoder, (f) constructing the decoder with an architecture symmetrical to the encoder, (g) constructing a second autoencoder with the encoder and the decoder, (h) training the second autoencoder, and (i) computing an error threshold.

[0017] Embodiments also include methods of feature/anomaly detection for cybersecurity. In aspects, the methods use a hybrid quantum-classical autoencoder.

[0018] Embodiments also include methods of detecting anomalies in cyber activity. In aspects, the anomalies are advanced persistent threats (APTs). In aspects, the anomalies are generated by botnets, ransomware, convergence, invisibility, and/or legacy/rogue devices.

[0019] The methods can include constructing a hybrid quantum-classical autoencoder using steps of (a) constructing an input for an input layer by concatenation of feature values of a sample, (b) training a first autoencoder to obtain trained weights corresponding to cybersecurity data, (c) constructing an encoder with the trained weights layers, (d) constructing and configuring a quantum circuit (parameters include qubits, number of layers of angle embedding and entangler), (e) adding the quantum circuit to the encoder, (f) constructing a decoder with an architecture symmetrical to the encoder, (g) constructing a second autoencoder with the encoder and the decoder and (h) training the second autoencoder.

[0020] In aspects, the method includes an additional step (i) of computing an error threshold for operational test sample analysis.

[0021] In aspects, the method can also include steps of determining a confidence score for an anomaly; determining that the confidence score is above a threshold associated with the cyberattack; and taking one or more remedial actions against the cyberattack.

[0022] In aspects, one or more remedial actions can be taken following detection of a cybersecurity attack.

[0023] In aspects, the embedding component of the quantum circuit is changed to another form of embedding (i.e., from angle embedding change to amplitude embedding/ IQP embedding).

[0024] In aspects, the number of layers in a quantum circuit is increased depending on the intended use. The number of qubits can also be adjusted to suit the expressivity of the quantum circuit.

[0025] In aspects, the quantum circuit can be run on a quantum device with the presence of quantum noise. In aspects, the quantum circuit can also be run on a quantum simulator that provides noiseless simulation with no source of error.

[0026] In aspects, the hybrid quantum-classical cyberattack detection application is executed in the cloud to detect a cyberattack. In aspects, software is hosted on the cloud and provided as a software as service (SaaS) service to a client.

### ***Definitions***

[0027] Reference in this specification to "one embodiment/aspect" or "an embodiment/aspect" means that a particular feature, structure, or characteristic described in connection with the embodiment/aspect is included in at least one embodiment/aspect of the disclosure. The use of the phrase "in one embodiment/aspect" or "in another embodiment/aspect" in various places in the specification are not necessarily all referring to the same embodiment/aspect, nor are separate or alternative embodiments/aspects mutually exclusive of other embodiments/aspects. Moreover, various features are described which may be exhibited by some embodiments/aspects and not by others. Similarly, various requirements are described which may be requirements for some embodiments/aspects but not other embodiments/aspects. Embodiment and aspect can in certain instances be used interchangeably.

[0028] The terms used in this specification generally have their ordinary meanings in the art, within the context of the disclosure, and in the specific context where each term is

used. Certain terms that are used to describe the disclosure are discussed below, or elsewhere in the specification, to provide additional guidance to the practitioner regarding the description of the disclosure. It will be appreciated that the same thing can be said in more than one way.

[0029] The term "advanced persistent threat" or "APT" refers to a stealthy threat actor, typically a state or state-sponsored group, which gains unauthorized access to a computer network and remains undetected for an extended period. In recent times, the term may also refer to non-state-sponsored groups conducting large-scale targeted intrusions for specific goals. Such threat actors' motivations are typically political or economic. Every major business sector has recorded instances of cyberattacks by advanced actors with specific goals, whether to steal, spy, or disrupt. These targeted sectors include government, defense, financial services, legal services, industrial, telecoms, consumer goods and many more. Some groups utilize traditional espionage vectors, including social engineering, human intelligence and infiltration to gain access to a physical location to enable network attacks. The purpose of these attacks is to install custom malware (i.e., malicious software).

[0030] The term "classifier" refers to the mathematical function, implemented by a classification algorithm that maps input data to a category. An algorithm that implements classification, especially in a concrete implementation, is known as a classifier.

[0031] The term "software as a service" or "SaaS" refers to a software licensing and delivery model in which software is licensed on a subscription basis and is centrally hosted. SaaS is also known as on-demand software, web-based software, or web-hosted software. SaaS is considered to be part of cloud computing, along with several other as a service business models. SaaS apps are typically accessed by users of a web browser (a thin client). SaaS became a common delivery model for many business applications, including office software, messaging software, development software, gamification, virtualization, etc.

[0032] The term “artificial intelligence” or “AI” refers to intelligence exhibited by machines, rather than humans. The term, as applied herein, refers to when a machine mimics “cognitive” functions that humans associate with other human minds, such as “learning” and “problem solving.”

[0033] The term “computer learning” or “machine learning” refers to an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it to learn for themselves.

[0034] The term “machine learning software” generally refers to a type of software application that uses artificial intelligence to make predictions or decisions based on data.

[0035] The term “deep learning” generally refers to a type of machine learning based on artificial neural networks in which multiple layers of processing are used to extract progressively higher-level features from data.

[0036] The term “module” such as that used in “deep learning module” generally refers to an extension to a software application’s main program that is dedicated to a specific function. For example, a “deep learning module” refers to an extension dedicated to deep learning which can perform all deep learning functions automatically without additional programming.

[0037] The term “neural network” refers to a method in artificial intelligence that teaches computers to process data in a way that is inspired by the human brain. It is a type of machine learning process, called deep learning, that uses interconnected nodes or neurons in a layered structure that resembles the human brain.

[0038] “Artificial neural networks” or “ANNs” are distributed computing systems that include a number of neurons interconnected through connection points called synapses.

Each synapse encodes the strength of the connection between the output of one neuron and the input of another. The output of each neuron is determined by the aggregate input received from other neurons that are connected to it. Thus, the output of a given neuron is based on the outputs of connected neurons from preceding layers and the strength of the connections as determined by the synaptic weights. An ANN is trained to solve a specific problem (e.g., pattern recognition) by adjusting the weights of the synapses such that a particular class of inputs produce a desired output.

[0039] Various algorithms can be used for this learning process. Certain algorithms may be suitable for specific tasks such as image recognition, speech recognition, or language processing. Training algorithms lead to a pattern of synaptic weights that, during the learning process, converges toward an optimal solution of the given problem.

[0040] Artificial neural networks include, for example, a feedforward neural network, a radial basis function network, a self-organizing map, learning vector quantization, a recurrent neural network, a Hopfield network, a Boltzmann machine, an echo state network, long short term memory, a bi-directional recurrent neural network, a hierarchical recurrent neural network, a stochastic neural network, a modular neural network, an associative neural network, a deep neural network, a deep belief network, a convolutional neural networks, a convolutional deep belief network, a large memory storage and retrieval neural network, a deep Boltzmann machine, a deep stacking network, a tensor deep stacking network, a spike and slab restricted Boltzmann machine, a compound hierarchical-deep model, a deep coding network, a multilayer kernel machine, or a deep Q-network.

[0041] A “Convolutional neural network” or “CNN” refers to a class of deep, feed-forward artificial neural networks, most commonly applied to analyzing visual imagery. CNNs use a variation of multilayer perceptrons designed to require minimal preprocessing. A CNN consists of an input and an output layer, as well as multiple hidden layers. The hidden layers of a CNN typically consist of convolutional layers, pooling layers, fully connected layers and normalization layers. Convolutional layers apply a convolution operation to

the input, passing the result to the next layer. Local or global pooling layers combine the outputs of neuron clusters at one layer into a single neuron in the next layer. Fully connected layers connect every neuron in one layer to every neuron in another layer. CNNs use relatively little pre-processing compared to other image classification algorithms. This means that the network learns the filters that in traditional algorithms were hand-engineered. This independence from prior knowledge and human effort in feature design is a major advantage.

[0042] The term “quantum gate” or “quantum logic gate” refers to a basic quantum circuit operating on a small number of qubits. By analogy to classical computing, quantum gates form quantum circuits, like classical logic gates form conventional digital circuits. Quantum logic gates are represented by unitary matrices. Various common quantum gates operate on spaces of one or two qubits, like classical logic gates operate on one or two bits. As matrices, quantum gates can be described by  $2^n \times 2^n$  sized unitary matrices, where  $n$  is the number of qubits. The variables that the gates act upon, the quantum states, are vectors in  $2^n$  complex dimensions. The base vectors indicate the possible outcomes if measured, and a quantum state is a linear combination of these outcomes. The action of the gate on a specific quantum state is found by multiplying the vector which represents the state by the matrix representing the gate. Accordingly, a given quantum state may be prepared on a quantum circuit through application of a plurality of gates. A given state may be characterized as a distribution function that provides a distribution describing a continuous random variable.

[0043] The term “quantum computing” refers to computing tasks or algorithms that can be accomplished by using quantum computers. With the rapid development of quantum technology, the tremendous potential of quantum computing makes this a key research direction in many countries. Considering the popularity of machine learning (ML), quantum machine learning (QML), which aims to combine the advantages of both quantum computing and machine learning, naturally receives considerable interest. To exploit the existing ML foundation for QML research, a promising direction is that of extending classical ML methods to their quantum or quantum-classical hybrid

counterparts. For example, QML models have been proposed such as the quantum autoencoder, quantum Boltzmann machine, quantum generation antagonistic learning and quantum kernel methods. Similar to the success of DNNs in classical machine learning, quantum deep neural networks (QDNNs) and quantum-classical hybrid deep neural networks (QHDNNs) also show extraordinary prospects in the field of QML.

[0044] The term “autoencoder” refers to a neural network that learns to compress data from the input layer into a short code, and then uncompress that code into something that closely matches the original data. This forces the autoencoder to engage in dimensionality reduction, for example by learning how to ignore noise. Autoencoders are also useful as generative models. An autoencoder can be used for anomaly detection. It learns a compact representation of healthy data. If there is a failure in detection of a particular data instance, there is a high probability that it encountered an anomaly. An autoencoder can be used for integration with a quantum circuit due to their bottleneck-like architecture. They compress their input, which enables passing high-dimensional classical data into small Noisy Intermediate-Scale Quantum (NISQ) computers.

[0045] A variational autoencoder or “VAE” refers to an artificial neural network architecture of the families of probabilistic graphical models and variational Bayesian methods. Variational autoencoders are often associated with the autoencoder model because of its architectural affinity, but with significant differences in the goal and mathematical formulation. Variational autoencoders are probabilistic generative models that require neural networks as only a part of their overall structure. The neural network components are typically referred to as the encoder and decoder for the first and second component respectively. The first neural network maps the input variable to a latent space that corresponds to the parameters of a variational distribution. In this way, the encoder can produce multiple different samples that all come from the same distribution. The decoder has the opposite function, which is to map from the latent space to the input space, in order to produce or generate data points. Both networks are typically trained together with the usage of the reparameterization trick, although the variance of the noise model can be learned separately.

[0046] The term “bottleneck” in a neural network refers to a layer with fewer neurons than the layer below or above it. Having such a layer encourages the network to compress feature representations (of salient features for the target variable) to best fit in the available space. Improvements to compression occur due to the goal of reducing the cost function, as for all weight updates. In a CNN, bottleneck layers are added to reduce the number of feature maps (aka channels) in the network, which, otherwise, tend to increase in each layer. This is achieved by using 1x1 convolutions with fewer output channels than input channels.

[0047] The term “remedial action” or “remediation” refers to addressing a breach or attack and limiting the amount of damage that breach can potentially cause to a network or business. Remedial actions can include but not limited to (a) containment, (b) eradication and (c) recovery. Containment generally involves taking steps to stop the breach from doing further damage to the network. After the threat has been contained, efforts may be directed at full remediation and complete removal of the threat from the system. This generally involves actively eradicating the threat itself (e.g., destroying malware, booting an unauthorized or rogue user from the network, etc.) and reviewing both affected and unaffected systems to ensure no traces of the breach are left behind. The final stage is recovery. When the threat has been entirely eradicated, affected systems can be restored to normal operations. This may involve deploying patches, rebuilding systems from backups, and bringing remediated systems and devices back online.

[0048] Other technical terms used herein have their ordinary meaning in the art that they are used, as exemplified by a variety of technical dictionaries. The particular values and configurations discussed in these non-limiting examples can be varied and are cited merely to illustrate at least one embodiment and are not intended to limit the scope thereof.

### BRIEF DESCRIPTION OF THE DRAWINGS

[0049] The accompanying drawings illustrate aspects of the present invention. In such drawings:

[0050] FIG. 1A depicts the basic scheme of a variational autoencoder according to embodiments.

[0051] FIG. 1B depicts the basic scheme of a hybrid quantum-classical autoencoder according to embodiments.

[0052] FIG. 2 depicts the architecture of a hybrid quantum autoencoder according to embodiments.

[0053] FIG. 3 is a flow chart of steps of constructing a hybrid quantum-classical neural network according to embodiments.

[0054] FIG. 4A – 4D are graphical depictions of results of validation studies shown as confusion matrices.

### DETAILED DESCRIPTION

[0055] It is to be understood that both the foregoing general description and the following detailed description are exemplary and explanatory and are intended to provide further explanation of the subject technology as claimed. Additional features and advantages of the subject technology are set forth in the description below, and in part will be apparent from the description, or may be learned by practice of the subject technology. The advantages of the subject technology will be realized and attained by the methods and systems particularly pointed out in the written description and claims hereof.

[0056] Anomaly detection (AD) refers generally to the task of identifying abnormal samples that differ from the given normal data. AD has important applications in many fields. For example, it can be used to monitor the operating states of machines or systems, detect whether systems are invaded, identify and detect financial fraud and perform out-of-distribution detection. The importance of AD has encouraged many researchers to develop AD algorithms. Due to the scarcity of anomalies, AD is often addressed as a task of semi-supervised/unsupervised learning. Its core idea is the assumption that the majority of accessible original data are normal, and the anomalous samples of data are not known a priori. Then, the goal is to build a mathematical model of normal data through training. When test data do not conform to the model, they are considered anomalous data.

[0057] There are many proposed solutions for AD including statistical methods, density-based methods, and distance-based methods. With the advent of deep learning and its success in recent machine learning (ML) research, AD methods based on deep neural networks (DNN) have attracted attention. Without a time-consuming feature engineering stage, deep AD makes it possible to learn a model directly from the raw input data, which can be more convenient and effective than the method of traditional AD. Therefore, such merits render deep AD particularly suitable for handling data which has grown due to the popularization of digital devices.

[0058] To improve the attack detection efficiency and accuracy, Applicants combined a quantum machine learning method (i.e., a quantum NN) with a classical NN.

#### ***Hybrid Quantum-classical Encoder***

[0059] The autoencoder learns a basic representation of the normal data and its reconstruction with minimum error. Therefore, the reconstruction error is used as an anomaly or classification metric. In addition to detecting anomaly data from normal data, the classification of anomaly types has also been investigated.

[0060] A classical computer works with ones and zeros whereas a quantum computer uses ones, zeros, and superpositions of ones and zeros. This enables quantum computers to perform a vast number of calculations simultaneously compared to classical computers. In a cloud-supported cyber-physical system environment, running a machine learning application in quantum computers is often difficult, due to the existing limitations of the current quantum devices. However, with the combination of quantum-classical neural networks (NN), complex and high-dimensional features can be extracted by the classical NN to a reduced but more informative feature space to be processed by the existing quantum computers.

[0061] In embodiments, a hybrid quantum-classical autoencoder (HAE) performs anomaly detection on classical data and is executable on Noisy Intermediate-Scale Quantum (NISQ) devices. The HAE can include a classical encoder, a parameterized quantum circuit (PQC) acting on its output and a classical decoder which then processes the measured output of the PQC.

[0062] The model is trained to reconstruct its input; and, therefore, it initially learns what is normal for the dataset under study. After the model is trained, the classical encoding layers and the PQC process the data, which is then analyzed by a standard outlier detector.

[0063] FIG. 1A depicts the basic scheme of a variational autoencoder (VAE) **100**. The model receives  $x$  as input **105**. The encoder **110** compresses it into the latent space **115**. The decoder **120** receives as input the information sampled from the latent space and produces  $x'$  as similar as possible to  $x$ .

[0064] FIG. 1B is a flow chart that shows the main components of a hybrid quantum-classical autoencoder **150**. In the first step **155**, data is compressed by the encoder (e.g., to a size of 5 units). This information is then embedded into the 5-qubit quantum circuit (QC) **160** by the encoding layer and processed by the processing layer. The circuit can have multiple layers. The final quantum state is then measured with respect to the

(standard) Z-basis, and a corresponding expectation value for each qubit is calculated. These expectation values span the latent space. Next, the decoder **165** reconstructs an input using the compressed data representation from the latent space. A reconstruction error is calculated based on the difference between the reconstructed input and the original input. Finally, the magnitude of the reconstruction error is compared against error threshold to detect if the original input is an anomaly (i.e., cyberattack).

[0065] FIG. 2 is a representation of an architecture of a hybrid quantum-classical autoencoder for anomaly detection. A quantum autoencoder combines the encoding layers of a variational autoencoder coupled with a quantum circuit as a bottleneck along with a decoder for training.

[0066] FIG. 3 is a block diagram **200** that shows steps of constructing a hybrid quantum-classical autoencoder for detecting an anomaly (i.e., a cyberattack) according to embodiments. The first step entails constructing an input for the input layer **205**. This can be achieved by concatenation of feature values of a sample or concatenation of feature values from sequential such as time series samples. Next step the first autoencoder **210** (e.g., variational autoencoder) is trained to obtain the trained weights corresponding to the cybersecurity data.

[0067] The encoder is constructed with the trained weights layers **215**. Then, the quantum circuit is constructed and configured **220**. Parameters can include qubits, number of layers of angle embedding and entangler. The quantum circuit is then added to the encoder **225**.

[0068] Next, the decoder is constructed with an architecture symmetrical to the encoder **230** and the second autoencoder is constructed with the encoder and decoder **235**. This step is depicted above in FIG. 2. The second autoencoder is then trained **240**. Finally, the error threshold for operational test samples analysis is computed using the configured reconstruction error **245**.

[0069] The number of layers in a quantum circuit can be increased depending on the users' needs. The number of qubits can also be adjusted to suit the expressivity of the quantum circuit.

### ***Operating Environment***

[0070] The machine learning software of the present invention is compatible with any operating system, including Mac OS, Microsoft Windows, Linux, Arthur, ARX, MOS, RISC iX, RISC OS, Fire OS, AmigaOS, Amiga Unix, AMSDOS, Contiki, CP/M 2.2, CP/M Plus, SymbOS, IBM, IBM AIX, Newton OS, iPadOS, watchOS, tvOS, bridgeOS, visionOS, XTS-400, BeOS, BeIA, Unix, MINI-UNIX, PWB/UNIX, CB UNIX, BESYS, Inferno, Burroughs MCP, GEOS, AmigaOS, AROS Research Operating System, SCOPE, Chippewa Operating System, MACE, Kronos, NOS, SIPROS, Puffin OS, CTOS, AOS, DG/UX, RDOS, CTOS, Deos, HeartOS, CP/M, Personal CP/M, CP/M Plus, CP/M-68K, CP/M-8000, CP/M-86 Plus, Personal CP/M-86, MP/M, MP/M II, FlexOS, Novell, or any other known operating system in the art.

[0071] The machine learning software of the present invention is also compatible with myriad graphical user interfaces. Compatible graphical user interfaces include those hosted on desktop computer monitors, laptop computer monitors, smartphone screens, smartwatch screens, television monitors, projection screens, multiplexed screens, LCD displays or any other hosted graphical user interface known in the art.

[0072] The invention may be described in the general context of computer-executable instructions, such as program modules, being executed by a computer. Generally, program modules include routines, programs, objects, components, data structures, etc., that perform particular tasks or implement particular abstract data types. The computer program and data may be fixed in any form (e.g., source code form, computer executable form, or an intermediate form) either permanently or transitorily in a tangible storage medium, such as a semiconductor memory device (e.g., a RAM, ROM, PROM, EEPROM, or Flash-Programmable RAM), a magnetic memory device (e.g., a diskette or fixed hard

disk), an optical memory device (e.g., a CD-ROM or DVD), a PC card (e.g., PCMCIA card), or other memory device. The computer program and data may be fixed in any form in a signal that is transmittable to a computer using any of various communication technologies, including, but in no way limited to, analog technologies, digital technologies, optical technologies, wireless technologies, networking technologies, and internetworking technologies. The computer program and data may be distributed in any form as a removable storage medium with accompanying printed or electronic documentation (e.g., shrink wrapped software or a magnetic tape), preloaded with a computer system (e.g., on system ROM or fixed disk), or distributed from a server or electronic bulletin board over the communication system (e.g., the Internet or World Wide Web). It is appreciated that any of the software components of the invention may, if desired, be implemented in ROM (read-only memory) form. The software components may, generally, be implemented in hardware, if desired, using conventional techniques.

[0073] The invention may also be practiced in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications network. In a distributed computing environment, program modules may be located in both local and remote computer storage media including memory storage devices. Practitioners of ordinary skill will recognize that the invention may be executed on one or more computer processors that are linked using a data network, including, for example, the Internet. In another embodiment, different steps of the process can be executed by one or more computers and storage devices geographically separated but connected by a data network in a manner so that they operate together to execute the process steps. In one embodiment, a user's computer can run an application that causes the user's computer to transmit a stream of one or more data packets across a data network to a second computer, referred to here as a server. The server, in turn, may be connected to one or more mass data storage devices where the database is stored. The server can execute a program that receives the transmitted packet and interpret the transmitted data packets in order to extract database query information. The server can then execute the remaining steps of the invention by means of accessing the mass storage devices to derive the desired result of the query. Alternatively, the server can

transmit the query information to another computer that is connected to the mass storage devices, and that computer can execute the invention to derive the desired result. The result can then be transmitted back to the user's computer by means of another stream of one or more data packets appropriately addressed to the user's computer. In one embodiment, the relational database may be housed in one or more operatively connected servers operatively connected to computer memory, for example, disk drives. In yet another embodiment, the initialization of the relational database may be prepared on the set of servers and the interaction with the user's computer occur at a different place in the overall process.

[0074] Further illustration of the present invention is shown in the working examples produced below.

## EXAMPLES

[0075] The following non-limiting examples are provided for illustrative purposes only in order to facilitate a more complete understanding of representative embodiments now contemplated. These examples are intended to be a mere subset of all possible contexts in which the components of the formulation may be combined. Thus, these examples should not be construed to limit any of the embodiments described in the present specification, including those pertaining to the type of components and/or methods and uses thereof.

### Validation of Software Model

[0076] Detecting anomalies in cyberactivity is crucial to effective cybersecurity. When analyzed over extended periods of time, web-based command and control traffic patterns exhibit distinctive network profiles, with the frequency and network profile being dependent on the specific threat, or malware family involved in the attack. The hybrid quantum-classical autoencoder described herein was validated to demonstrate its effectiveness.

**Results**

[0077] A confusion matrix is a tabular way of visualizing the performance of a prediction model. Each entry in a confusion matrix denotes the number of predictions made by the model where it classified the classes correctly or incorrectly.

[0078] A “true positive” or “TP” refers to the number of predictions where the classifier correctly predicts the positive class as positive. A “true negative” or “TN” refers to the number of predictions where the classifier correctly predicts the negative class as negative. Similarly, a “false positive” or “FP” refers to the number of predictions where the classifier incorrectly predicts the negative class as positive. A “false negative” or “FN” refers to the number of predictions where the classifier incorrectly predicts the positive class as negative.

[0079] FIG. 4A shows the results of a validation test as a confusion matrix. The system recognized 74.38% of the anomalies (i.e., true positives). Only 5.06% of anomalies were mis-identified (i.e., false positives).

[0080] The validation test was repeated with a second data set. FIG. 4B shows the results of a validation test as a confusion matrix. The system recognized 98.86% of the anomalies (i.e., true positives). Only 14.78% of anomalies were mis-identified (i.e., false positives).

[0081] FIG. 4C shows the results of a validation test as a confusion matrix. The system recognized 88.32% of the anomalies (i.e., true positives). Only 9.44% of anomalies were mis-identified (i.e., false positives).

[0082] The validation test was repeated with a second data set. FIG. 4D shows the results of a validation test as a confusion matrix. The system recognized 98.43% of the anomalies (i.e., true positives). Only 12.06% of anomalies were mis-identified (i.e., false positives).

[0083] Based on these results, it is clear that the assessed AI algorithm is accurate for recognizing anomalies among large amounts of data. The assessed AI model can improve identification and allow remedial actions to improve cybersecurity and prevent attacks.

\* \* \*

[0084] In closing, it is to be understood that although aspects of the present specification are highlighted by referring to specific embodiments, one skilled in the art will readily appreciate that these disclosed embodiments are only illustrative of the principles of the subject matter disclosed herein. Therefore, it should be understood that the disclosed subject matter is in no way limited to a particular compound, composition, article, apparatus, methodology, protocol, and/or reagent, etc., described herein, unless expressly stated as such. In addition, those of ordinary skill in the art will recognize that certain changes, modifications, permutations, alterations, additions, subtractions and sub-combinations thereof can be made in accordance with the teachings herein without departing from the spirit of the present specification. It is therefore intended that the following appended claims and claims hereafter introduced are interpreted to include all such changes, modifications, permutations, alterations, additions, subtractions and sub-combinations as are within their true spirit and scope.

[0085] Certain embodiments of the present invention are described herein, including the best mode known to the inventors for carrying out the invention. Of course, variations on these described embodiments will become apparent to those of ordinary skill in the art upon reading the foregoing description. The inventor expects skilled artisans to employ such variations as appropriate, and the inventors intend for the present invention to be practiced otherwise than specifically described herein. Accordingly, this invention includes all modifications and equivalents of the subject matter recited in the claims appended hereto as permitted by applicable law. Moreover, any combination of the above-described embodiments in all possible variations thereof is encompassed by the invention unless otherwise indicated herein or otherwise clearly contradicted by context.

[0086] Groupings of alternative embodiments, elements, or steps of the present invention are not to be construed as limitations. Each group member may be referred to and claimed individually or in any combination with other group members disclosed herein. It is anticipated that one or more members of a group may be included in, or deleted from, a group for reasons of convenience and/or patentability. When any such inclusion or deletion occurs, the specification is deemed to contain the group as modified thus fulfilling the written description of all Markush groups used in the appended claims.

[0087] Unless otherwise indicated, all numbers expressing a characteristic, item, quantity, parameter, property, term, and so forth used in the present specification and claims are to be understood as being modified in all instances by the term "about." As used herein, the term "about" means that the characteristic, item, quantity, parameter, property, or term so qualified encompasses a range of plus or minus ten percent above and below the value of the stated characteristic, item, quantity, parameter, property, or term. Accordingly, unless indicated to the contrary, the numerical parameters set forth in the specification and attached claims are approximations that may vary. For instance, as mass spectrometry instruments can vary slightly in determining the mass of a given analyte, the term "about" in the context of the mass of an ion or the mass/charge ratio of an ion refers to  $\pm 0.50$  atomic mass unit. At the very least, and not as an attempt to limit the application of the doctrine of equivalents to the scope of the claims, each numerical indication should at least be construed in light of the number of reported significant digits and by applying ordinary rounding techniques.

[0088] Use of the terms "may" or "can" in reference to an embodiment or aspect of an embodiment also carries with it the alternative meaning of "may not" or "cannot." As such, if the present specification discloses that an embodiment or an aspect of an embodiment may be or can be included as part of the inventive subject matter, then the negative limitation or exclusionary proviso is also explicitly meant, meaning that an embodiment or an aspect of an embodiment may not be or cannot be included as part of the inventive subject matter. In a similar manner, use of the term "optionally" in reference to an

embodiment or aspect of an embodiment means that such embodiment or aspect of the embodiment may be included as part of the inventive subject matter or may not be included as part of the inventive subject matter. Whether such a negative limitation or exclusionary proviso applies will be based on whether the negative limitation or exclusionary proviso is recited in the claimed subject matter. Further, the use of the terms “include,” “includes” and “including” means include, includes and or including as well as include, includes and including, but not limited to.

[0089] Notwithstanding that the numerical ranges and values setting forth the broad scope of the invention are approximations, the numerical ranges and values set forth in the specific examples are reported as precisely as possible. Any numerical range or value, however, inherently contains certain errors necessarily resulting from the standard deviation found in their respective testing measurements. Recitation of numerical ranges of values herein is merely intended to serve as a shorthand method of referring individually to each separate numerical value falling within the range. Unless otherwise indicated herein, each individual value of a numerical range is incorporated into the present specification as if it were individually recited herein.

[0090] All patents, patent publications, and other publications referenced and identified in the present specification are individually and expressly incorporated herein by reference in their entirety for the purpose of describing and disclosing, for example, the compositions and methodologies described in such publications that might be used in connection with the present invention. These publications are provided solely for their disclosure prior to the filing date of the present application. Nothing in this regard should be construed as an admission that the inventors are not entitled to antedate such disclosure by virtue of prior invention or for any other reason. All statements as to the date or representation as to the contents of these documents is based on the information available to the applicants and does not constitute any admission as to the correctness of the dates or contents of these documents.

[0091] The flowchart and block diagrams in the figures illustrate the architecture, functionality, and operation of possible implementations of systems, methods and

computer program products according to various embodiments of the present invention. In this regard, each block in the flowchart or block diagrams may represent a module, segment, or portion of code, which comprises one or more executable instructions for implementing the specified logical function(s). It should also be noted that, in some alternative implementations, the functions noted in the block may occur out of the order noted in the figures. For example, two blocks shown in succession may, in fact, be executed substantially concurrently, or the blocks may sometimes be executed in the reverse order or out of order, depending upon the functionality involved. It will also be noted that each block of the block diagrams and/or flowchart illustration, and combinations of blocks in the block diagrams and/or flowchart illustration, can be implemented by special purpose hardware-based systems that perform the specified functions or acts, or combinations of special purpose hardware and computer instructions.

[0092] Lastly, the terminology used herein is for the purpose of describing particular embodiments only and is not intended to limit the scope of the present invention, which is defined solely by the claims. Accordingly, the present invention is not limited to that precisely as shown and described.

[0093] Although embodiments of the current disclosure have been described comprehensively in considerable detail to cover the possible aspects, those skilled in the art would recognize that other versions of the disclosure are also possible.

[0094] While the present invention has been described in terms of particular embodiments and applications, in both summarized and detailed forms, it is not intended that these descriptions in any way limit its scope to any such embodiments and applications, and it will be understood that many substitutions, changes and variations in the described embodiments, applications and details of the method and system illustrated herein and of their operation can be made by those skilled in the art without departing from the spirit of this invention.

## CLAIMS

What is Claimed is:

1. A method of detecting a cyberattack, the method comprising steps of:
  - (a) constructing an input for an input layer by concatenation of feature values from a sample,
  - (b) training a first autoencoder to obtain trained weights corresponding to cybersecurity data,
  - (c) constructing an encoder using the trained weights,
  - (d) constructing and configuring a quantum circuit,
  - (e) adding the quantum circuit to the encoder,
  - (f) constructing a decoder with an architecture symmetrical to the encoder,
  - (g) constructing a second autoencoder with the encoder and the decoder,
  - (h) training the second autoencoder, and
  - (i) computing an error threshold.
2. The method of claim 1, further comprising a step of identifying a cyberattack and taking one or more remedial actions against the cyberattack.
3. The method of claim 1, further comprising steps of:
  - (j) identifying a cyberattack by determining that a data point is outside the error threshold; and
  - (k) taking one or more remedial actions against the cyberattack.
4. The method of claim 1, wherein the first autoencoder is a variational autoencoder.
5. The method of claim 1, wherein the step of constructing and configuring a quantum circuit comprises setting parameters, the parameters selected from a number of qubits, a number of layers of angle embedding and/or use of an entangler.

6. A hybrid quantum-classical autoencoder comprising:
- (a) a classical computing component, and
  - (b) a quantum computing component,
- wherein the hybrid quantum-classical autoencoder is configured to combine encoding layers of a variational autoencoder coupled with a quantum circuit as a bottleneck with a decoder for training, and
- wherein the hybrid quantum-classical computer is configured to detect an anomaly from a data source.
7. A method of detecting a cyberattack using the hybrid quantum-classical autoencoder of claim 6.
8. The method of claim 7, further comprising a step of taking one or more remedial actions against the cyberattack.
9. The hybrid quantum-classical autoencoder of claim 6, wherein the data source is comprised of network data.
10. A method of detecting an anomaly among a population of data points, the method comprising steps of:
- (a) constructing an input for an input layer by concatenation of feature values from a sample,
  - (b) training a first autoencoder to obtain trained weights,
  - (c) constructing an encoder with the trained weights,
  - (d) constructing and configuring a quantum circuit,
  - (e) adding the quantum circuit to the encoder,
  - (f) constructing a decoder with an architecture symmetrical to the encoder,
  - (g) constructing a second autoencoder with the encoder and the decoder,
  - (h) training the second autoencoder, and
  - (i) computing an error threshold.

11. The method of claim 10, further comprising steps of identifying the anomaly as a cyberattack and taking one or more remedial actions against the cyberattack.
12. The method of claim 10, further comprising steps of:
  - (j) identifying the anomaly by determining a data point is outside the error threshold,
  - (k) confirming that the anomaly is a cyberattack, and
  - (l) taking one or more remedial actions against the cyberattack.
13. The method of claim 10, wherein the first autoencoder is a variational autoencoder.
14. The method of claim 10, wherein the step of constructing and configuring a quantum circuit comprises setting parameters, the parameters selected from a number of qubits, a number of layers of angle embedding and/or use of an entangler.
15. A system for detecting anomalies for cybersecurity, the system comprised of a first autoencoder that uses classical computing, a quantum circuit and a decoder that uses quantum computing.
  - wherein the system is configured to combine encoding layers of autoencoder coupled with the quantum circuit as a bottleneck with a decoder for training, and
  - wherein the system is configured to detect anomalies from a data source.
16. The system of claim 15, wherein the system is constructed by:
  - (a) providing an input for an input layer by concatenation of feature values from a sample,
  - (b) training the first autoencoder to obtain trained weights corresponding to cybersecurity data,
  - (c) constructing the encoder with the trained weights layers,
  - (d) constructing and configuring the quantum circuit,
  - (e) adding the quantum circuit to the encoder,

- (f) constructing the decoder with an architecture symmetrical to the encoder,
- (g) constructing a second autoencoder with the encoder and the decoder,
- (h) training the second autoencoder, and
- (i) computing an error threshold.

17. The system of claim 16, wherein the first autoencoder is a variational autoencoder.
18. The system of claim 16, wherein the step of constructing and configuring a quantum circuit comprises setting parameters, the parameters selected from a number of qubits, a number of layers of angle embedding and/or use of an entangler.