

SYSTEM AND METHOD FOR DATA AUTONOMY VIA INFERENCE PATHWAYS IN MISSION ORCHESTRATION PLATFORM

TECHNICAL FIELD

[0001] This disclosure pertains to a cutting-edge mission orchestration platform designed to address deficiencies in mission data management and processing, ensuring data autonomy and integrity. It involves the novel use of multi-layered inference pathway protocol for the autonomous arbitration of artificial intelligence (AI) models to manage and process mission data. This advanced approach of deploying multi-layered inference pathway protocols facilitates extremely fast data self-organization, enhances data independence in mission platforms, maintains high data integrity, and accelerates the extraction of actionable intelligence from diverse data types sourced in mission orchestration.

BACKGROUND

[0002] In the domain of mission orchestration platforms, especially those utilizing modern databases such as, but not limited to, Hadoop, MongoDB, ElasticSearch or Cassandra, there are numerous operational challenges, primarily due to the management of large volumes of heterogeneous data types such as video, images, text, sound, and other non-standardized data types. The labor-intensive nature of running and maintaining these systems is significant, as it involves comprehensive management of diverse data stores and sophisticated data retrieval operations. Additionally, during periods of high data influx, bottlenecks may occur in the system as the rapid intake of data may saturate the system's capacity. Tools capable of handling a wide array of data formats—structured, unstructured, semi-structured, and reference data—are usually employed to detect patterns that may indicate anomalies and/or events. Challenges associated with managing such extensive data collections stem from the sheer volume, diversity, and the speed at which the data must be processed.

[0003] For example, once data patterns and/or insights have been successfully identified in the mission orchestration platforms, the respective notifications will then be generated for the human operators by these platforms. However, the volume of notifications will often surpass what human operators can manage effectively and efficiently. The processes involved in analyzing these notifications typically necessitate detailed examination and may involve the analysis of specific information about the subjects and any particular directives for handling the content of the notifications. Moreover, human operators usually spend a

significant amount of time and effort to fully understand the nature and implications of detected events, often leading to severe fatigue and a high attrition rate.

[0004] In particular, the cognitive demands placed on human operators by these tasks are considerable. The primary issue stems from the sheer volume and rapid growth of such data, which existing mission orchestration platforms tend to struggle to accommodate efficiently. This not only necessitates scalable storage solutions but also robust mechanisms for ensuring data integrity and redundancy, significantly escalating infrastructure costs and complexity. From an analytical perspective, extracting actionable insights from the data that triggered the notifications requires advanced analytical tools and considerable computational resources. Most contemporary implementations tend to slow down or become unusable, or even crash when the system's entropy reach critical levels.

[0005] In response to the challenges associated with managing high volumes and varieties of data in mission orchestration platforms, several methods have been proposed by those skilled in the art.

[0006] Among the methods proposed thus far, the application of AI models stands out as a particularly effective alternative to traditional human-operated manual analysis and triage of the notifications content within the mission orchestration platforms. While the foundational concept of artificial intelligence—which comprise the steps of providing computers with training data to autonomously develop predictive models—is straightforward, its practical implementation presents substantial challenges. For example, multiple versions and/or types of AI models end up being designed for similar tasks, and each model tends to utilize different technologies, programming languages, and libraries. These variations present a hurdle in ensuring compatibility across these AI models. Furthermore, the performance of these models may vary considerably based on the types of input data they process.

[0007] Existing solutions that enhance mission orchestration platforms inundated with diverse data streams typically involve the direct application of AI models with minimal preprocessing to adapt the data into a compatible format for the models. These models, which are tasked with performing inference based on the data received, usually comprise complex algorithms and are resource intensive. Therefore, there is an ongoing pursuit by industry experts for more adaptive and autonomous learning systems that can perform both atomic and

macro correlations of data efficiently. These autonomous systems tend to provide more effective solutions to the complex challenges prevalent in modern data management and analysis operations.

SUMMARY

[0008] The present disclosure aims to transform the technology used to handle large volumes of data in mission orchestration platforms by employing a novel inference pathway protocol to autonomously manage the analysis of these data groups. The primary objective is to diminish human involvement by empowering machines to effectively review and address data patterns and insights generated from the analysis of the data, thereby decreasing the necessity for continuous human oversight and constant supervision.

[0009] The technology proposed here automates the intricate processes of data correlation, analysis, and triage, dramatically improving operational efficiency and accuracy, reducing human error, and decreasing dependency on large-scale human oversight.

[0010] This technology introduces a novel approach for integrating multiple artificial intelligence (AI) models within a cohesive framework in mission orchestration platforms. This modular approach not only facilitates seamless processing and autonomous decision-making across various data types but also ensures dynamic adaptability to both structured and data environments. Furthermore, each module is capable of autonomously adjusting its operational parameters to optimize its processing methodology for the specific type of data it encounters, whether it is streaming real-time video or batch-processed textual data.

[0011] At the heart of the system's architecture lies the implementation of sophisticated AI models designed to supplant traditional manual monitoring methods. These models are continuously fed with a diverse range of data inputs, enabling them to iteratively learn and enhance their predictive capabilities. To maintain high levels of accuracy and performance, the platform is equipped with mechanisms that ensure compatibility and interoperability among various models, which may utilize different technologies, programming languages, and libraries. This interoperability is crucial for maintaining system robustness and agility, accommodating the rapid evolution of data processing technologies.

[0012] In an aspect of the present disclosure, a computing module for autonomously reviewing and triaging data in a database is disclosed. The disclosed computing module comprises a processing unit and a non-transitory media readable by the processing unit. The media stores instructions that when executed by the processing unit causes the processing unit to retrieve data from the database, process the data using a feature extraction model to identify a plurality of features within the data and assign a rank to each of the identified features based on a predefined ranking criterion. For each feature, in descending order based on its assigned rank, the processing unit selects an inference pathway protocol from a set of inference pathway protocols using a first classification model selector, whereby the first classification model selector was trained based on a set of rules associated with the feature characteristics and data type and classifies the feature using the selected inference pathway protocol. Each inference pathway protocol comprises a hierarchical, multi-layered pathway structure including a base node for receiving inputs, and a plurality of classification models arranged as path nodes at each layer of the pathway structure, such that each path node processes the input data and branches into further pathways until reaching terminal nodes configured with multi-class classification models that produce the final classification outputs. The processing unit then aggregates the final classification outputs generated by the inference pathway protocols, selects, using a second classification model selector, a trained artificial intelligence (AI) model from a set of AI models stored in a database, the selection being based on the aggregated final classification outputs and executes an inference task using the selected AI model, the inference task being based on the aggregated final classification output. The processing unit then performs an action on the data based on an outcome of the inference task.

[0013] In another aspect of the disclosure, a method for autonomously reviewing and triaging data in a database using a computing module is disclosed, the method comprising the steps of retrieving an data from the database, identifying features in the data using a feature extraction model, and ranking each of the identified features. In this method, for each feature in the data and in a descending ranking order based on a corresponding ranking of each feature, an inference pathway protocol is selected from a set of inference pathway protocols using a first classification model selector, and the feature is classified using the selected inference pathway protocol, whereby the first classification model selector was trained based on a set of rules, and whereby each inference pathway protocol comprises a plurality of classification models arranged in a multi-layered pathway structure with inputs to each of the

inference pathway protocols being provided to a base node of the corresponding pathway structure, and final classification outputs being generated by multi-class classification models provided at terminal nodes of the corresponding pathway structure, wherein each layer of the pathway structure has a path node comprising a classification model. The method then selects, using a second classification model selector, a trained AI model from a set of trained AI models based on a combination of the final classification outputs generated by all the inference pathway protocols, performing an inference task based on the combination of the classifications and then performs an action on the data based on an outcome of the inference task.

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] Various embodiments of the present disclosure are described below with reference to the following drawings:

Figure 1 illustrates a block diagram of a system for autonomously reviewing and triaging data in a database of a system in accordance with embodiments of the present disclosure;

Figure 2 illustrates a block diagram of a computing module for autonomously reviewing and processing groups of data in a database in accordance with embodiments of the present disclosure;

Figure 3 illustrates a block diagram representative of a processing system for performing embodiments of the present disclosure; and

Figure 4 illustrates a flowchart of a process for autonomously reviewing processing groups of data in a system in accordance with embodiments of the present disclosure.

DETAILED DESCRIPTION

[0015] The following detailed description is made with reference to the accompanying drawings, showing details and embodiments of the present disclosure for the purposes of illustration. Features that are described in the context of an embodiment may correspondingly be applicable to the same or similar features in the other embodiments, even if not explicitly described in these other embodiments. Additions and/or combinations and/or alternatives as described for a feature in the context of an embodiment may correspondingly be applicable to the same or similar feature in the other embodiments.

[0016] In the context of various embodiments, the articles “a”, “an” and “the” as used with regard to a feature or element include a reference to one or more of the features or elements.

[0017] In the context of various embodiments, the term “about” or “approximately” as applied to a numeric value encompasses the exact value and a reasonable variance as generally understood in the relevant technical field, e.g., within 10% of the specified value.

[0018] As used herein, the term “and/or” includes any and all combinations of one or more of the associated listed items.

[0019] As used herein, “comprising” means including, but not limited to, whatever follows the word “comprising”. Thus, use of the term “comprising” indicates that the listed elements are required or mandatory, but that other elements are optional and may or may not be present.

[0020] As used herein, “consisting of” means including, and limited to, whatever follows the phrase “consisting of”. Thus, use of the phrase “consisting of” indicates that the listed elements are required or mandatory, and that no other elements may be present.

[0021] As used herein, “data” means including, but not limited to, various data formats such as structured, unstructured, semi-structured, and reference data.

[0022] As used herein, when certain components, modules or the like are described as “configured to” perform certain functions, it is to be understood that such a configuration may be accomplished, by example, by an electronic circuit or hardware that is designed to perform the function and this may be through software programming or by controlling the functions of electronic circuits in a particular manner to perform the function or combinations of the functions.

[0023] Further, one skilled in the art will recognize that certain functional units in this description have been labelled as modules throughout the specification. The person skilled in the art will also recognize that a module may be implemented as circuits, logic chips or any

sort of discrete component. Still further, one skilled in the art will also recognize that a module may be implemented in software which may then be executed by a variety of processor architectures. In embodiments of the disclosure, a module may also comprise computer instructions or executable code that may instruct a computer processor to carry out a sequence of events based on instructions received. The choice of the implementation of the modules is left as a design choice to a person skilled in the art and does not limit the scope of the claimed subject matter in any way.

[0024] Further, one skilled in the art will also recognize that the detailed workings, internal structures, techniques and training datasets of AI models, neural networks and anomaly assertion models referred to in this disclosure have not been disclosed or shown in detail as such information is known to one skilled in the art and as such, has been omitted for brevity.

[0025] Artificial intelligence (AI) models, such as but not limited to convolutional neural networks (CNN), deep neural networks (DNN), recurrent neural networks (RNN), or other types of AI models, are developed through a structured training process. This process involves providing a learning algorithm—the mechanism by which the model learns—with training data from which to learn. The training data usually incorporates the correct responses, known as targets or target attributes, often referred to as the "ground truth." During training, the learning algorithm provides this data to the AI model, which produces a corresponding output. The algorithm then assesses the performance of the AI model by calculating the loss or error relative to the correct answers and subsequently adjusts the model's parameters to minimize this loss or error. Essentially, the learning algorithm identifies patterns in the training data that correlate input data attributes with the target, resulting in a trained AI model that encapsulates these identified patterns.

[0026] In a mission orchestration platform that handles and processes large datasets, critical operations such as quick responses, data analysis, and monitoring are heavily reliant on the effective management of diverse and voluminous data from a variety of sources. These sources tend to generate a significant amount of data and these data may comprise videos, audio files, temperature measurements, pressure measurements, olfactory measurements, images, text data, and non-text data, where all these various types of data files usually

comprise multiple formats. In embodiments of the disclosure, video data may be obtained using image capturing devices such as cameras configured to capture and record live video feeds; audio data may be obtained using audio capturing devices such as microphones and audio sensors configured to capture live audio feeds, audio recordings, or ambient sound data; temperature and pressure data may be obtained by measurement devices such as temperature or pressure sensors configured to provide real-time temperature or pressure readings; and text data may be obtained from various sources such as reports, incident logs, text messages, and emails. One skilled in the art will recognize that other types of data may be captured and recorded using various other means.

[0027] In particular, video streams offer a visual perspective on various areas, while audio feeds provide auditory cues and context. Simultaneously, temperature and pressure sensors deliver crucial environmental data. Additionally, text data sources offer insights and reports relevant to particular matters. The mission orchestration platform's robust configuration enables seamless integration and analysis of all these data streams, providing an all-encompassing view of the ecosystem to ensure that any potential anomalies or irregularities do not go unnoticed.

[0028] To handle this plethora of information from disparate sources, data collected from these sources are aggregated into groups of data. Each group stores data as single entities within a database system, thereby accommodating the storage of large elements of data, such as multimedia files, without the need for the data to conform to the rigid types required by traditional database fields.

[0029] In embodiments of the disclosure, instead of storing the groups of data directly in a database system, the groups of data are temporarily stored in a cache, such as a memory buffer. The caching of the data serves several functions: it enhances the speed of data retrieval and processing, allowing for quicker operational responses; it reduces the load on the storage system; and it prevents the redundant processing of the same data multiple times.

[0030] Figure 1 illustrates a block diagram of a system for autonomously reviewing and triaging data in accordance with embodiments of the present disclosure. System 100 comprises computing module 102 that is configured to process data 101 before the processed

outcome is provided to data management system 114. In embodiments of the disclosure, computing module 102 comprises database 103, feature extraction module 104, ranking module 106, classification model selector module 108, and databases 110 and 112.

[0031] In embodiments of the disclosure, database 103 may comprise a cache that is configured to store groups of data 101. The cache may comprise of a high-speed memory component and it is usually smaller and faster than the main memory of computing module 102 and operates at speeds closer to the processing speed of the processor of computing module 102, thereby reducing latency and improving the overall performance of the computing module. Hence, the cache may comprise a Static Random-Access Memory (SRAM) or a Dynamic Random Access Memory (DRAM), depending on the application's cost and performance requirements

[0032] In embodiments of the disclosure, feature extraction module 104 is configured to select a feature extraction model from an ensemble of feature extraction models that are stored within a database of module 104. Each of these models are configured to extract features from various types of media, such as, but not limited to, images, videos, audio clips, and text logs. The ensemble may include AI models like convolutional neural networks (CNNs) that may be used to identify objects, patterns, and activities in image and video data, recurrent neural networks (RNNs) and their variants such as Long Short-Term Memory (LSTM) networks that are used for analyzing audio clips for speech recognition or anomaly detection and natural language processing (NLP) models for processing and extracting meaningful information from text logs, reports, and emails. The detailed workings of these feature extraction AI models are omitted for brevity as they are well known in the art.

[0033] Depending on the format and/or type of surveillance data that is to be processed by the computing module, feature extraction module 104 will autonomously select the appropriate feature extraction model to be applied to the data. This selection may be based on a predefined rules-based approach, which could involve predefined criteria such as data type (i.e., characteristics of the data), source, or specific analytical needs. Details of the implementation of such a rules-based approach are omitted for brevity as they are known to one skilled in the art.

[0034] In a furtherance of the embodiment described above, when a data block retrieved from database 103 comprises an image file, feature extraction module 104 may retrieve a Convolutional Neural Network (CNN) type of feature extraction model from the database of feature extraction module 104 as CNNs are typically used to automatically detect important features without any human supervision. In a typical CNN, the input image passes through a series of convolutional layers that apply various filters (kernels) to detect edges, colors, textures, and other spatial hierarchies of features. These features are then pooled (usually via max pooling) to reduce dimensionality and passed through fully connected layers to classify or encode the image into a feature vector. This vector captures the essential characteristics of the image and can be used for further processing or analysis. In embodiments of the disclosure, a You-Only-Look-Once (YOLO) object detection model which frames object detection as a single regression problem may also be used or a Single Shot Multi-box Detector (SSD) model may also be used to detect and extract features in the image.

[0035] In further embodiments, when the input data comprises a video file, feature extraction module 104 may retrieve from the database in module 104 a feature extraction model that comprises a CNN architecture (as described above) that has been further extended to encompass 3D Convolutional Neural Networks, CNNs adapted for spatial feature extraction, Recurrent Neural Networks (RNN) or Long Short-Term Memory networks (LSTMs). 3D CNNs are usually utilized in situations where processing multiple frames simultaneously is necessary to capture both spatial and temporal features, thereby recognizing patterns over time as well as space. Alternatively, a CNN can process individual frames to extract spatial features, followed by the utilization of an RNN or LSTM to analyze the sequence of CNN-generated feature vectors, thereby capturing dynamics and movements over time.

[0036] In a further embodiment of the disclosure, when the input data comprises an audio file, feature extraction module 104 may retrieve from the database a feature extraction model based on a Mel-frequency Cepstral Coefficients (MFCCs) or a Spectrogram-based CNN. MFCCs are a type of feature extraction model that effectively represents the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear Mel scale of frequency. MFCCs are widely used in speech and audio processing because they mimic the human auditory system's response more closely than the linearly

spaced frequency bands used in the normal Fourier transform. Alternatively, the audio files may be transformed into a spectrogram and feature extraction models comprising the previously described CNNs may be used to extract the features from spectrograms, similar to how image data files are processed, extracting robust features that capture both spectral and temporal characteristics.

[0037] In embodiments of the disclosure, ranking module 106 may comprise a ranking model that is configured to sort items, features or entities based on their relevance or importance relative to a specific query or task. A basic ranking model comprises a scoring function that is configured to assign a numerical score to each feature based on its importance or relevance and this function may comprise, but is not limited to, a linear combination of weighted features or a deep neural network trained to process input features in a non-linear manner. In general, ranking models may be categorized into three types based on how they evaluate relevance between items: Pointwise, Pairwise, and Listwise approaches. Pointwise models treat the ranking problem as a regression or classification task, predicting the relevance score or classifying the relevance of individual items independently. Pairwise models, such as RankNet, focus on correctly ordering pairs of items by learning from comparisons; they predict whether one item in a pair is more relevant than the other and are commonly used in settings like search engine result ordering. Listwise models consider the entire list of items and aim to optimize the order of all items collectively; examples include ListNet and LambdaRank, which are particularly effective in scenarios where the overall sequence of items impacts performance, such as in search engine rankings or recommended lists. One skilled in the art will recognize that the choice of the ranking model is dependent on the feature that is to be ranked.

[0038] In embodiments of the disclosure, classification model selector module 108 may comprise a variety of selector models and/or benchmarking models, which are instrumental in managing and optimizing the selection and deployment of AI models within a model group or inference pathway protocols within another model group. Each selector model may be configured to utilize various strategies to select one or more models from the model group to process specific input data. This selection can be random, sequential (such as round-robin), rules-based or based on more complex rules like frequency-based partitioning for A/B testing, attribute-based partitioning, or multi-armed bandit strategies. Additionally, the

selector module may be configured to refine its selection rules dynamically based on previous performance data. This dynamic selection is often guided by specific business objectives or performance indicators, such as based on feedback from a human operator or accuracy rates, which serve as scoring metrics to assess the effectiveness of the models being deployed. The detailed operation of a classification model selector is omitted for brevity as it is known to one skilled in the art.

[0039] In further embodiments of the disclosure, classification model selector module 108 may be provided with an evaluator model. The evaluator model may be configured to assess various models offline, during the phase before the selection of the model by the selector module. The evaluator model achieves this by scoring the outputs produced by the selected AI models against a test dataset using accuracy metrics such as confusion matrices and the area under a Receiver Operation Characteristic (ROC) curve. This evaluation helps in comparing different models or different versions of the same model to determine which one performs best under given conditions. The evaluation process can also include visual tools provided by the artificial intelligence platform to aid in decision-making. The results produced by the evaluator module may then be used to influence the selection process of the selector model.

[0040] In embodiments of the disclosure, database 110 may comprise a plurality of inference pathway protocols. It is useful at this stage to note that in most applications such as in object identification and classification use cases, there may be a need to classify inputs into a large number of classes, potentially exceeding tens of thousands of classes. Training a single classification model to handle such a vast array of classes can pose significant challenges, in addition to the usual difficulties in achieving convergence, excessive training times, and high demands on computational resources and memory. Additionally, even if training is technically feasible, the practicality of using a single model for such extensive classification or inference tasks is often questionable due to these constraints.

[0041] Hence, to address these challenges, in embodiments of this disclosure, inference pathway protocols or models may be used in place of conventional large complex AI models. Inference pathway protocols employ multiple classification models organized into several layers within an inference pathway structure with inputs being provided to the base node of

the pathway structure and with the final output being generated after the input has been selectively passed through a series of branching pathways. Each node within a branch of this multi-layered pathway structure comprises a distinct classification model (i.e., a binary classification model), which could be any type from linear classifiers like logistic regression or naive Bayes, to more complex models such as support vector machines (SVMs), decision trees, nearest neighbor classifiers, neural networks or large language models. These models can be independently selected, trained, and updated according to the needs at each node on the branching pathways. Typically, except at the terminal nodes (which are the final nodes at the bottom of the inference pathway structure), each node operates as a binary classification model, dividing the inputs into two categories or groups that ideally, but not necessarily, contain an equal number of classes.

[0042] The inference pathway structure allows for the configuration of the depth of the paths (i.e., the number of branches of paths) and the number of classes each path node handles, with limits often set to make training more manageable and efficient (e.g., fewer than 50 layers, with each path node handling fewer than 20 classes in each group of classes). Presently, there is a myriad of classification models where each model is trained for the classification of a specialized type of object and/or feature. Without the instantiation of an inference pathway protocol as described above, the efficient application of such models becomes implausible. The inference pathway framework facilitates the judicious selection and application of appropriate classification models to the objects under scrutiny. Consequently, this framework enables parallel processing across all pathways, commencing with the identification of the highest-ranked object.

[0043] The incorporation of this inference pathway protocol not only streamlines the classification process but also optimizes computational resources by enabling concurrent processing of multiple pathways. By systematically prioritizing the identification of objects based on their ranking, the efficiency of the classification system is markedly enhanced, thereby underscoring the significance of this model within the context of object classification methodologies.

[0044] In other words, such a structure ensures that the models at the path nodes are easier to train and that the overall system is not overly deep, allowing for rapid and predictable

classification processes. This multi-layered pathway approach not only enhances the manageability of training with numerous classes but also improves the speed and accuracy of the classification task, making it highly suitable for applications requiring the differentiation of many classes, such as in systems designed to understand and respond to varied user intents based on their input messages.

[0045] In most implementations of inference pathway protocols, the structure of the pathways of each of the inference pathway protocols is established prior to the selection and training of the classification models at each node. This process typically begins by categorizing all available classes into two approximately equal groups. These groups are then assigned to two child nodes at a subsequent layer of the pathway model. This splitting process is repeated at each layer, continuously dividing the classes into nearly equal subsets until the number of classes at each node falls below a predetermined threshold.

[0046] This method effectively creates a near-binary pathway structure. At each non-terminal node in this pathway structure, a binary classification model is selected and trained. These binary classifiers are advantageous as they can be trained more swiftly and consume less memory compared to models handling larger sets of classes. This streamlined approach facilitates a rapid and efficient training process, crucial for managing the broad and complex data sets typically encountered in such systems.

[0047] At the terminal points of the pathway structure—the terminal nodes—a multiclass classification model is implemented. The use of multiclass models at these points is strategic as it limits the overall complexity or depth of the pathway structure, ensuring that the classification process remains efficient even as it reaches the finer granularity required at these terminal nodes. This setup allows the system to handle a diverse array of classes while maintaining manageable computational demands. Moreover, the models at each node of the tree can be trained either in parallel or sequentially.

[0048] When an inference pathway structure is queried, the input to the structure is provided to the base node. From this base node, the input will then be processed by the respective path nodes (i.e., the classification models) and will be guided through the various layers based on the logic defined at each node, until the relevant terminal nodes or desired

depth is reached. This approach allows for efficient data filtering and extraction, as each node serves as a decision point that progressively refines the search or decision process, directing the query path towards the most relevant subset of data.

[0049] The effectiveness of querying an inference pathway structure is significantly enhanced by the structure's ability to compartmentalize and organize data in a multi-layered approach, which inherently reduces the search space and improves query response times. The multilayered nature allows for logical, stepwise refinement in the query process, ensuring that each step is focused and based on the accumulated criteria from previous nodes, thus optimizing the querying process for accuracy and efficiency.

[0050] In embodiments of the disclosure, database 112 may comprise AI models that have been trained to perform higher-level classifications based on a plurality of inputs. Specifically, these trained AI models are configured to determine if a plurality of features provided to the model signifies an event or an anomaly in the data, such as unauthorized access, intrusion, or any other anomaly. These trained AI models may include, but are not limited to Support Vector Machines (SVM), CNNs, Random Forests, Neural Networks, Anomaly Detection models and etc. In the context of identifying abnormal events, SVMs can classify complex features extracted from images into categories like "normal" and "anomaly" by finding the hyperplane that best separates the different classes with a maximum margin while CNNs can be trained on labeled data of both normal and compromising incidents to learn distinguishing features directly. Random Forest models are particularly useful in contexts where decisions need to be made reliably under varied and possibly noisy conditions. The ensemble approach helps in handling overfitting and provides a practical way to assess the importance of the different features for classification while Neural Network models can learn from large amounts of labeled data (images tagged as normal or indicative of an event) to discern subtle patterns that might indicate an abnormality. Lastly, in situations where it is not feasible to label a vast array of images, anomaly detection models can be used. These models learn what "normal" combinations of features are like and then flags deviations from this norm as potential events. The anomaly detection models may employ autoencoders, which are a type of neural network, to reconstruct normal events and highlight anomalies by detecting reconstruction errors.

[0051] With reference to Figure 2, in operation, when a continuous input of data 101 is provided to computing module 102, data 101 is converted into groups of data such as groups 101a-101d which are then stored in database 103 as groups of data 203a-203n. In embodiments of the disclosure, each group of data may comprise data having the same format and/or type or may be grouped according to data markers used to signify the beginning of a sequence of received data. The groups of data may then be retrieved in a sequential manner or based on a ruleset by feature extraction module 104. Feature extraction module 104 will then autonomously select the appropriate feature selection model that is to be applied to each retrieved group of data based on the format and/or type of the data.

[0052] Once the features of a group of data (e.g., data 203a) have been identified by module 104, these identified features will then be ranked by ranking module 106 based on predefined ranking criterion. Module 106 is configured to sort items, features or entities based on their relevance or importance relative to a specific query or task. For example, if module 106 is ranking objects/features generated from an image file, it may rank the features according to their size or position in the image, if it is ranking features from a video file, in addition to the size or position of the features, it may rank the features according to the frequency of occurrence of a feature over a few video frames, and if it ranking features from an audio file, it may rank the features according to the decibel of a feature, or the frequency of occurrence of the feature over a timeframe. One skilled in the art will recognize that these are just some examples and that module 106 may utilize other parameters that may be determined by one skilled in the art to rank the features of the groups of data.

[0053] In embodiments of the disclosure, the ranking module may comprise a Convolutional Neural Network (CNN) as such neural networks are highly effective for tasks that involve the ranking of images as they have an inherent ability to autonomously extract and learn features directly from image data. These networks are structured with convolutional layers that apply filters to the inputs, capturing essential features such as edges, textures, and more complex patterns at deeper layers. This capability makes CNNs particularly suited for analyzing visual content in images, which are crucial for ranking them based on various criteria such as aesthetic quality, size, relevance, or specific content. The training process involves using a large dataset of pre-ranked or scored images, where the network learns to correlate specific visual features with their corresponding ranks through a series of training

and validation steps to adjust the network's parameters effectively. Once trained, the CNN can then be deployed to perform image ranking tasks dynamically.

[0054] Computing module 102 will then proceed to classify each of the features identified by module 104 based on the rank assigned to each of these features. In embodiments of the disclosure, features having the more important rank (i.e., those having a higher-ranking score) will be classified first and more computing resources may be allocated to this higher ranked object as required. One skilled in the art will recognize that the order may be switched or varied, and such a variation in the processing order is still covered in this disclosure.

[0055] During the classification of each of the features, computing module 102 may utilize classification model selector module 108 to select an inference pathway protocol from database 110 that is to be used to classify a feature selected from the plurality of identified features. For example, if five unique different features are to be classified, selector module 108 may select up to five different types of inference pathway protocols from database 110, with each model specifically deployed to classify one of these unique features.

[0056] After all the identified features of the group of data have been classified by their respective inference pathway protocols, all these features are then provided to classification model selector module 108. Classification model selector module 108 proceeds to select an AI model from database 112 that is best suited to perform an inference task based on the features contained in this group of data. Computing module 102 then performs the action based on the outcome generated by the selected trained AI model and the processed group of data is then provided to data management system 114. This process is repeated in the next group of data (i.e., on data 203b) until all the groups of data have been processed as described above.

[0057] It should be noted that such an approach is advantageous as in a scenario whereby multiple objects of different types are contained within a group of data; data autonomy could be utilized by deploying specialized sub-models that independently focus on particular types of features within the group of data. For instance, if the group of data represents an image file, a first type of inference pathway protocol might be used for recognizing vehicles, while another type of inference pathway protocol might be better suited for detecting humans.

These models operate autonomously, applying their specialized knowledge only to relevant parts of the image, and then the results are aggregated at a higher level using the more complex AI models contained in database 112 to produce a cohesive output.

[0058] In summary, such an approach allows for inference pathway autonomy as each segment of a group of data can be processed separately to identify unique attributes, which are then integrated in higher layers of the model to form a unified understanding of the group of data. This method is particularly useful in groups of data which contain densely packed information where different objects might overlap or interact in a manner that is hidden from the human eye.

[0059] In another embodiment of the disclosure whereby the data comprises an image file, before the step of providing the identified features that have been classified by their respective inference pathway protocols to classification model selector module 108, each of the classified objects from this image file may also be tagged with the location of the classified object within the image file.

[0060] Once this is done, a graph representation of these classified objects may be constructed where each classified object corresponds to a node in the graph. Each of these nodes may encapsulate various attributes of the object, including dimensions of the object, coordinates, and any other features extracted by the inference pathway used to classify the object. For example, if the image file contained five unique objects that were classified in the previous step, each of these classified objects is then represented as a node in the graph.

[0061] In the graph, edges between nodes are established based on spatial relationships between each of the classified objects. For instance, if two objects are sufficiently close to each other or if their bounding boxes overlap, a connection or edge is formed between them to denote their spatial relationship. These edges encode information about the proximity and relative positions of objects within the image. For example, edges might connect two objects in an image that are adjacent to each other or in close proximity.

[0062] A graph neural network (GNN) architecture may be obtained from database 112 and then be used to process this graph representation of detected objects. The GNN takes

node features (object attributes) and edge features (spatial relationships) as input, propagating information between nodes and edges to capture spatial dependencies. During inference, the trained GNN model predicts spatial relationships between objects based on their features and contextual information encoded in the graph structure. For example, a message passing GNN architecture may be configured to iteratively update node representations based on information received from neighboring nodes, allowing it to infer complex spatial relationships between objects. The GNN used in this embodiment may be trained based on previously generated annotated data where the spatial relationships between various types of objects have been defined. The detailed working of the GNN is omitted for brevity as the structure of standard GNNs are well known to one skilled in the art.

[0063] The previously identified features that have been classified by their respective inference pathway protocols together with the predicted spatial relationships between the classified objects may then be provided to classification model selector module 108. Classification model selector module 108 then proceeds to select an AI model from database 112 that is able to carry out or execute an actionable inference task based on the input provided. Computing module 102 then performs the action based on the outcome generated by the selected trained AI model.

[0064] The integration of the final classification outputs as obtained from the inference pathway protocols with spatial relationships between classified objects presents a substantial advancement in artificial intelligence methodologies. Inference pathway protocols are adept at decomposing complex classification challenges into more manageable segments, enabling precise identification of objects across various categories. By further combining these detailed classification outputs with spatial relationship data, the resultant AI model gains the capacity to not only discern the identity of objects but also to comprehend their mutual interactions or relative positions within a given space. This dual-layered approach significantly enhances both the accuracy and the contextual depth of the analysis. By understanding the context of how objects relate to one another spatially, the model can make informed predictions and decisions that consider the dynamic environment of the objects.

[0065] Additionally, the capability to recognize complex patterns that emerge from both the classification of objects and their spatial relationships enables the machine model to

undertake more sophisticated analysis as this pattern recognition would extend beyond simple object identification, encompassing the analysis of object groupings, movement patterns over time, and spatial distributions. Furthermore, the inclusion of spatial relationships helps in reducing biases inherent in models trained primarily on appearance-based features. These biases can lead to misclassifications or erroneous predictions when models rely solely on visual identifiers without considering the contextual significance of objects within their environment. By integrating spatial context, the model's dependence on potentially biased visual features is diminished, enhancing its generalizability and accuracy.

[0066] In yet another embodiment of the disclosure, before the step of providing the identified features that have been classified by their respective inference pathway protocols to classification model selector module 108, computing module 102 may perform a chaining of the pathways across the various inference pathway protocols to identify any pathways from the plurality of inference pathway protocols that may lead to the same final classification output for different features. The chaining of the pathways generally involves the identification of the pathways within different inference pathway protocols that lead to the same final classification outcome. Once the converging pathways of the different inference pathway protocols have been identified, computing module 102 then aggregates information from each path of each different inference pathway protocol to identify features or attributes that influenced the classification decisions along the node of each path along the pathway. Computing module 102 then analyses the consistency of the classification decisions across these multiple paths to generate a confidence value in the final classification results. If the aggregation of information across the various pathways show significant consistency in the reasoning or feature importance used in the various AI models at each of the path nodes, this implies a strong likelihood that the classifications were performed correctly and that the results are reliable. The features that have been classified by their respective inference pathway protocols may then be provided to classification model selector module 108 to be actioned upon as previously described above.

[0067] Conversely, if the paths show significant discrepancies in their reasoning or feature importance, it may indicate potential weaknesses or ambiguities in the classification criteria used in the respective pathways. The previously obtained classification results of the associated features may then be discarded and computing module 102 may trigger

classification model selector module 108 to select other inference pathway protocols from database 110 to perform another round of classification on the affected features. This feedback loop, which is triggered by the chaining of the pathways across the various inference pathway protocols ensures that new variations or complexities in the data may be handled more accurately and efficiently. The features that have been classified by their respective inference pathway protocols may then be provided to classification model selector module 108 to be actioned upon as previously described above.

[0068] In accordance with embodiments of the present disclosure, a block diagram representative of components of processing system 300 that may be provided within computing module 102 or any of the modules provided within computing module 102 to carry out any of the functions described above is illustrated in Figure 3. One skilled in the art will recognize that the exact configuration of each processing system provided within these modules may be different and the exact configuration of processing system 300 may vary and the arrangement illustrated in Figure 3 is provided by way of example only.

[0069] In embodiments of the invention, processing system 300 may comprise controller 301 and user interface 302. User interface 302 is arranged to enable manual interactions between a user and the computing module as required and for this purpose includes the input/output components required for the user to enter instructions to provide updates to each of these modules. A person skilled in the art will recognize that components of user interface 302 may vary from embodiment to embodiment but will typically include one or more of display 340, keyboard 335 and optical device 336.

[0070] Controller 301 is in data communication with user interface 302 via bus 315 and includes memory 320, processor 305 mounted on a circuit board that processes instructions and data for performing the method of this embodiment, an operating system 306, an input/output (I/O) interface 330 for communicating with user interface 302 and a communications interface, in this embodiment in the form of a network card 350. Network card 350 may, for example, be utilized to send data from these modules via a wired or wireless network to other processing devices or to receive data via the wired or wireless network. Wireless networks that may be utilized by network card 350 include, but are not limited to, Wireless-Fidelity (Wi-Fi), Bluetooth, Near Field Communication (NFC), cellular

networks, satellite networks, telecommunication networks, Wide Area Networks (WAN) and etc.

[0071] Memory 320 and operating system 306 are in data communication with CPU 305 via bus 310. The memory components include both volatile and non-volatile memory and more than one of each type of memory, including Random Access Memory (RAM) 323, Read Only Memory (ROM) 325 and a mass storage device 345, the last comprising one or more solid-state drives (SSDs). One skilled in the art will recognize that the memory components described above comprise non-transitory computer-readable media and shall be taken to comprise all computer-readable media except for a transitory, propagating signal. Typically, the instructions are stored as program code in the memory components but can also be hardwired. Memory 320 may include a kernel and/or programming module such as a software application that may be stored in either volatile or non-volatile memory.

[0072] Herein the term “processor” is used to refer generically to any device or component that can process such instructions and may include: a microprocessor, microcontroller, programmable logic device or other computational device. That is, processor 305 may be provided by any suitable logic circuitry for receiving inputs, processing them in accordance with instructions stored in memory and generating outputs (for example to the memory components or on display 340). In this embodiment, processor 305 may be a single core or multi-core processor with memory addressable space. In one example, processor 305 may be multi-core, comprising—for example—an 8 core CPU. In another example, it could be a cluster of CPU cores operating in parallel to accelerate computations.

[0073] A process for identifying anomaly constructs in real-time based on surveillance data is illustrated in Figure 4 whereby process 400 may be carried out by a computing module in conjunction with modules contained within the computing module in accordance with embodiments of the disclosure.

[0074] Process 400 begins at step 402 with process 400 retrieving data from a database. Process 400 then proceeds to identify features in the data using a feature extraction model at step 402. At step 404, process 400 then identifies features in the data using a feature extraction model. Rankings are then generated for each of the identified features by process

400 and this takes place at step 406. A feature having the highest rank is then selected at step 408 by process 400. Process 400 then selects an inference pathway protocol based on the selected feature from a set of inference pathway protocols using a first classification model selector and proceeds to classify the selected feature at step 410.

[0075] If process 400 determines at step 412 that there is another feature associated with the data that has not yet been classified, process 400 then proceeds to step 414 to select the next feature with the next highest rank. Process 400 then proceeds to step 410 where process 400 selects an inference pathway protocol based on the selected feature from a set of inference pathway protocols using the first classification model selector and proceeds to classify the selected feature at step 410.

[0076] In embodiments of the disclosure, the first classification model selector may be trained based on a set of rules, and each inference pathway protocol may comprise a plurality of classification models arranged in a multi-layered pathway structure with inputs to each of the inference pathway protocols being provided to a base node of the corresponding pathway structure, and final classification outputs being generated by multi-class classification models provided at terminal nodes of the corresponding pathway structure. Further, each layer of the pathway structure was provided with a path node which may comprise of a binary classification model.

[0077] As an example, when the feature to be classified is provided to a base node of the inference pathway protocol, the base node which comprises a binary classification model will then evaluate if the feature should be aligned with a first set of classes or a second set of classes. The outcome of this evaluation may then be expressed in terms of a result comprising of two probabilities – representing the likelihood of the feature belonging to either set of classes. The process then moves to another path node associated with the class that had the higher probability values (e.g. the first set of classes) whereby this path node is located on another layer in the inference pathway structure. At the other path node, another binary classification model is used to evaluate if the input feature should be aligned with a first or a second subset of classes of the first set of classes. This process then progresses through the various layers of the inference pathway structure until it arrives at a terminal node which then proceeds to generate the final classification output.

[0078] If process 400 determines at step 412 that there is yet another feature associated with the data that has not yet been classified, process 400 then proceeds to step 414. Steps 414 to 412 repeat itself until all the features identified in step 404 have been classified.

[0079] Process 400 then proceeds to step 416 whereby process 400 selects, using a second classification model selector, a trained AI model from a set of trained AI models based on a combination of the final classification outputs generated at step 410. Process 400 then performs an inference task based on the combination of the classification outputs. At step 418, process 400 then performs an action on the data based on the outcome at step 416 and attaches this result to the data for further processing.

[0080] In an embodiment of the disclosure, each of the classification models of the path nodes in each of the sets of inference pathway protocols are trained to classify inputs into two groups of classes and the multi-class classification models are trained to classify inputs into a single class.

[0081] In an embodiment of the disclosure, the retrieved data may comprise an image file, and before the step of selecting a trained AI model from a set of trained AI models, process 400 may proceed to determine spatial relationships between all the final classification outputs as generated by the inference pathway protocols.

[0082] In a further embodiment, process 400 may use the determined spatial relationships between all the final classification outputs together with the final classification outputs generated by all the inference pathway protocols to select the trained AI model from a set of trained AI models.

[0083] In another embodiment, as process 400 is classifying the features using the selected inference pathway protocols, process 400 may perform a chaining of pathways on inference pathway protocols that generated final classification outputs that are similar. Process 400 may then proceed to discard the generated final classification outputs that are similar when it is determined that from the chaining results that discrepancies exist in the chained pathways. Process 400 then proceeds to trigger the first classification model selector to select another

inference pathway protocol from the set of inference pathway protocols to classify the features associated with the discarded final classification outputs.

[0084] In another embodiment of the disclosure, the classification model at the path nodes may comprise a support vector machine classifier, a neural network classifier, a nearest neighbor classifier, a decision tree classifier, a logistic regression classifier, or a naïve Bayes classifier and the multi-class classification models may comprise a neural network classifier, a support vector machine classifier, a K-nearest neighbor classifier or a large language model. Process 400 may also be configured to perform the classification of each of the features of the data in parallel.

[0085] Numerous other changes, substitutions, variations, and modifications may be ascertained by the skilled in the art and it is intended that the present application encompass all such changes, substitutions, variations, and modifications as falling within the scope of the appended claims.

CLAIMS:

1. A computing module for autonomously reviewing and triaging data in a database, the computing module comprising:
 - a processing unit; and
 - a non-transitory media readable by the processing unit, the media storing instructions that when executed by the processing unit causes the processing unit to:
 - retrieve data from the database;
 - process the data using feature extraction to identify a plurality of features within the data;
 - assign a rank to each of the identified features based on a predefined ranking criterion;
 - for each feature based on its assigned rank, select an inference pathway protocol from a set of inference pathway protocols using a first classification model selector, whereby the first classification model selector was trained based on a set of rules associated with the feature characteristics and data type, and
 - classify the feature using the selected inference pathway protocol, whereby each inference pathway protocol comprises:
 - a hierarchical, multi-layered pathway structure including a base node for receiving inputs, and
 - a plurality of classification models arranged as path nodes at each layer of the pathway structure, such that each path node processes the input data and branches into further pathways until reaching terminal nodes configured with multi-class classification models that produce the final classification outputs;
 - aggregate the final classification outputs generated by the inference pathway protocols;
 - select, using a second classification model selector, a trained artificial intelligence (AI) model from a set of AI models stored in a database, the selection being based on the aggregated final classification outputs;
 - execute an inference task using the selected AI model, the inference task being based on the aggregated final classification output; and
 - perform an action on the data based on an outcome of the inference task.
2. The computing module according to claim 1 wherein each of the plurality of classification models arranged as the path nodes in the set of inference pathway protocols

were trained to classify inputs into two groups of classes and wherein the multi-class classification models were trained to classify inputs into a single class.

3. The computing module according to claim 1 wherein the retrieved data comprises an image file, and wherein before the instructions to select a trained AI model from a set of trained AI models, the computing module further comprises instructions that direct the processing unit to:

determine spatial relationships between all the final classification outputs as generated by the inference pathway protocols and provide the spatial relationships to the second classification model selector.

4. The computing module according to claim 3 wherein the instructions to select the trained AI model from a set of trained AI models further comprises instructions for directing the processing unit to:

use the determined spatial relationships between all the final classification outputs together with the final classification outputs generated by all the inference pathway protocols to select the trained AI model from a set of trained AI models.

5. The computing module according to claim 1 wherein the instructions to classify the feature using the selected inference pathway protocol further comprises instructions for directing the processing unit to:

perform a chaining of pathways on inference pathway protocols that generated final classification outputs that are similar.

6. The computing module according to claim 5 wherein the instructions to perform the chaining of pathways further comprises instructions for directing the processing unit to:

discard the generated final classification outputs that are similar when it is determined that from the chaining results that discrepancies exist in the chained pathways; and

trigger the first classification model selector to select another inference pathway protocol from the set of inference pathway protocols to classify the features associated with the discarded final classification outputs.

7. The computing module according to claim 1 wherein the plurality of classification models arranged as the path nodes may each comprise a support vector machine classifier, a

neural network classifier, a nearest neighbour classifier, a decision tree classifier, a logistic regression classifier, a naïve Bayes classifier, or a large language model.

8. The computing module according to claim 1 wherein the multi-class classification models may comprise a neural network classifier, a support vector machine classifier, a K-nearest neighbour classifier, or a large language model.
9. The computing module according to claim 1 wherein the processing unit performs the classification of each of the features of the data occurs in parallel.
10. A method for autonomously reviewing and triaging data in a database using a computing module comprises:
 - retrieving data from the database;
 - processing the data using a feature extraction model to identify a plurality of features within the data;
 - assigning a rank to each of the identified features based on a predefined ranking criterion;
 - for each feature, selecting an inference pathway protocol from a set of inference pathway protocols using a first classification model selector, whereby the first classification model selector was trained based on a set of rules associated with the feature characteristics and data type, and
 - classifying the feature using the selected inference pathway protocol, whereby each inference pathway protocol comprises:
 - a hierarchical, multi-layered pathway structure including a base node for receiving inputs, and
 - a plurality of classification models arranged as path nodes at each layer of the pathway structure, such that each path node processes the input data and branches into further pathways until reaching terminal nodes configured with multi-class classification models that produce the final classification outputs;
 - aggregating the final classification outputs generated by the inference pathway protocols;
 - selecting, using a second classification model selector, a trained artificial intelligence (AI) model from a set of AI models stored in a database, the selection being based on the aggregated final classification outputs;

executing an inference task using the selected AI model, the inference task being based on the aggregated final classification output; and
performing an action on the data based on an outcome of the inference task.

11. The method according to claim 10 wherein each of the plurality of classification models arranged as the path nodes in the set of inference pathway protocols were trained to classify inputs into two groups of classes and wherein the multi-class classification models were trained to classify inputs into a single class.

12. The method according to claim 10 wherein the retrieved data comprises an image file, and wherein before the step of selecting a trained AI model from a set of trained AI models, the method further comprises the steps of:

determining spatial relationships between all the final classification outputs as generated by the inference pathway protocols and providing the spatial relationships to the second classification model selector.

13. The method according to claim 12 wherein the step of selecting the trained AI model from a set of trained AI models further comprises the steps of:

using the determined spatial relationships between all the final classification outputs together with the final classification outputs generated by all the inference pathway protocols to select the trained AI model from a set of trained AI models.

14. The method according to claim 10 wherein the step of classifying the feature using the selected inference pathway protocol further comprises the steps of:

performing a chaining of pathways on inference pathway protocols that generated final classification outputs that are similar.

15. The method according to claim 14 wherein the step of performing the chaining of pathways further comprises the steps of:

discarding the generated final classification outputs that are similar when it is determined that from the chaining results that discrepancies exist in the chained pathways;
and

triggering the first classification model selector to select another inference pathway protocol from the set of inference pathway protocols to classify the features associated with the discarded final classification outputs.

16. The method according to claim 10 wherein the plurality of classification models arranged as the path nodes may each comprise a support vector machine classifier, a neural network classifier, a nearest neighbour classifier, a decision tree classifier, a logistic regression classifier, a naïve Bayes classifier or a large language model.
17. The method according to claim 10 wherein the multi-class classification models may comprise a neural network classifier, a support vector machine classifier, a K-nearest neighbour classifier or a large language model.
18. The method according to claim 10 wherein the classification of each of the features of the data occurs in parallel.