CLAIMS

What is claimed is:

1. A method for identifying a cause of data disparity among monitored data, the method comprising:

monitoring, using a machine learning model executing on a server computer, parameters associated with data collected in connection with a processing computer;

detecting, using the machine learning model, a data disparity among the data;

identifying, using the machine learning model, a first event associated with the data disparity;

determining, using the machine learning model, a set of events associated with the first event;

calculating, using the machine learning model, a score for each event among the set of events as a function of a weight assigned to each event among the set of events and occurrence score determined for each event among the set of events;

identifying, using the machine learning model, a second event among the set of events as the cause of the data disparity, wherein the second event has the highest score among the set of events;

when a third event, different than the second event, is an actual cause of the data disparity, training the machine learning model by:

adding the third event to the set of events if the third event is not included in the set of events;

adjusting the weight of each event among the set of events by a predetermined amount, wherein adjusting includes increasing the weight of the third event; and

recalculating the score for each event among the set of events; and

taking, using the machine learning model, preventive actions to prevent the data disparity in view of the identified cause of the data disparity.

2. The method of claim 1, wherein the occurrence score of a selected event is determined based on runtime characteristics of all child events and parent events of the selected event, wherein the selected event occurred prior to the all child events of the selected event, and all parent events of the selected event occurred prior to the selected event.

3. The method of claim 1, further comprising two or more iterations of adjusting and recalculating, wherein the score of the third event increases at each iteration such that the third event has the highest score among the set of events at conclusion of all iterations.

4. The method of claim 1, wherein the weight of a given event is stored along with a history of the given event being the actual cause of the data disparity.

5. The method of claim 1, further comprising:

receiving an alert associated with the data disparity;

in response to the alert, identifying the first event associated with the data disparity.

6. The method of claim 1, wherein the set of events associated with the first event includes one or more parent events of the first event, wherein the one or more parent events occurred prior to the first event.

7. The method of claim 1, wherein the first event and the set of events form a dependency graph.

8. The method of claim 7, further comprising:

adding a new event to the dependency graph, wherein the new event is associated with a fourth event and a fifth event in the set of events, wherein the fifth event is a descendent of the fourth event;

associating the new event with the fourth event without associating with the fifth event.

9. The method of claim 1, wherein the first event and the set of events include one or more of an internal event to the processing computer and an external event to the processing computer.

10. A computer comprising:

a processor; and

a computer readable medium, the computer readable medium comprising code that, when executed by the processor, cause the processor to:

monitor, using a machine learning model executing on the processor, parameters associated with data collected in connection with a processing computer;

detect, using the machine learning model, a data disparity among the data;

identify, using the machine learning model, a first event associated with the data disparity;

determine, using the machine learning model, a set of events associated with the first event;

calculate, using the machine learning model, a score for each event among the set of events as a function of a weight assigned to each event among the set of events and occurrence score determined for each event among the set of events;

identify, using the machine learning model, a second event among the set of events as the cause of the data disparity, wherein the second event has the highest score among the set of events;

when a third event, different than the second event, is an actual cause of the data disparity, training the machine learning model by:

adding the third event to the set of events if the third event is not included in the set of events;

adjusting the weight of each event among the set of events by a predetermined amount, wherein adjusting includes increasing the weight of the third event; and

recalculating the score for each event among the set of events; and

take, using the machine learning model, preventive actions to prevent the data disparity in view of the identified cause of the data disparity.

11. The computer of claim 10, wherein the occurrence score of a selected event is determined based on runtime characteristics of all child events and parent events of the selected event, wherein the selected event occurred prior to the all child events of the selected event, and all parent events of the selected event occurred prior to the selected event.

12. The computer of claim 10, wherein the weight of a given event is stored along with a history of the given event being the actual cause of the data disparity.

13. The computer of claim 10, wherein the code, when executed by the processor, further causes the processor to:

receive an alert associated with the data disparity;

in response to the alert, identify the first event associated with the data disparity.

14. The computer of claim 10, wherein the set of events associated with the first event includes one or more parent events of the first event, wherein the one or more parent events occurred prior to the first event.

15. The computer of claim 10, wherein the first event and the set of events form a dependency graph.

16. The computer of claim 15, wherein the code, when executed by the processor, further causes the processor to:

add a new event to the dependency graph, wherein the new event is associated with a fourth event and a fifth event in the set of events, wherein the fifth event is a descendent of the fourth event;

associate the new event with the fourth event without associating with the fifth event.

17. The computer of claim 10, wherein the first event and the set of events include one or more of an internal event to the processing computer and an external event to the processing computer.