

WHAT IS CLAIMED IS:

1. A computer-implemented method comprising:

obtaining, with at least one processor, first training data associated with a first set of features and second training data associated with a second set of features different than the first set of features, wherein each of the first set of features and the second set of features is associated with a same plurality of transactions and a same plurality of labels for the same plurality of transactions;

training, with at least one processor, a first model based on the first training data, the second training data, and the same plurality of labels; and

training, with at least one processor, a second model, using a loss function that depends on an output of an intermediate layer of the first model and an output of the second model, based on the second training data,

wherein the second model includes at least one first layer and at least one second layer, wherein the output of the second model includes an output of the at least one first layer, and wherein training the second model further comprises:

providing, with the at least one processor, as input to the second model, only the second training data;

receiving, with the at least one processor, as output from the second model, the output of the second model;

modifying, using the loss function that depends on the output of the intermediate layer of the first model and the output of the second model including the output of the at least one first layer, one or more parameters of the at least one first layer of the second model to learn to match the output of the intermediate layer of the first model; and

training the at least one second layer based on the output of the at least one first layer and the same plurality of labels,

wherein the first model includes a recurrent neural network that includes two long short-term memory (LSTM) hidden layers including the intermediate layer of the first model, wherein the output of the intermediate layer of the first model includes a hidden state vector $h(t)$ from the second LSTM hidden layer of the two LSTM hidden layers, wherein the second model includes a feedforward regression neural network (FRNN), and wherein the loss function minimizes a squared error (L2) loss between the output of the second model and the output of the intermediate layer of the first model.

2. The computer-implemented method of claim 1, wherein the first model includes at least one of the following: a deep neural network, a recurrent neural network, an ensemble of a plurality of neural networks, or any combination

thereof, wherein the first layer of the second model includes a regression neural network, and wherein the second layer of the second model includes a logistic regression model.

3. The computer-implemented method of claim 1, further comprising:

determining, with at least one processor, a plurality of information values of a plurality of intermediate layers of the first model; and

selecting, with at least one processor, the intermediate layer from the plurality of intermediate layers based on the plurality of information values.

4. The computer-implemented method of claim 1, wherein the first set of features includes complex features, and wherein the second set of features includes interpretable features.

5. The computer-implemented method of claim 1, wherein the first model includes a greater number of parameters than the second model.

6. The computer-implemented method of claim 1, further comprising:

providing, with at least one processor, the trained second model;

obtaining, with at least one processor, input data associated with at least one transaction; and

processing, with at least one processor and using the trained second model, the input data to generate output data, wherein the output data includes a prediction of whether the at least one transaction is a fraudulent transaction.

7. A computing system comprising:

at least one processor programmed and/or configured to:

obtain first training data associated with a first set of features and second training data associated with a second set of features different than the first set of features, wherein each of the first set of features and the second set of features is associated with a same plurality of transactions and a same plurality of labels for the same plurality of transactions;

train a first model based on the first training data, the second training data, and the same plurality of labels; and

train a second model, using a loss function that depends on an output of an intermediate layer of the first model and an output of the second model, based on the second training data,

wherein the second model includes at least one first layer and at least one second layer, wherein the output of the second model includes an output of the at least one first layer, and wherein the at least one processor further trains the second model by:

providing, as input to the second model, only the second training data;

receiving, as output from the second model, the output of the second model;

modifying, using the loss function that depends on the output of the intermediate layer of the first model and the output of the second model including the output of the at least one first layer, one or more parameters of the at least one first layer of the second model to learn to match the output of the intermediate layer of the first model; and

training the at least one second layer based on the output of the at least one first layer and the same plurality of labels,

wherein the first model includes a recurrent neural network that includes two long short-term memory (LSTM) hidden layers including the intermediate layer of the first model, wherein the output of the intermediate layer of the first model includes a hidden state vector $h(t)$ from the second LSTM hidden layer of the two LSTM hidden layers, wherein the second model includes a feedforward regression neural network (FRNN), and wherein the loss function minimizes a squared error (L2) loss between the output of the second model and the output of the intermediate layer of the first model.

8. The computing system of claim 7, wherein the first model includes at least one of the following: a deep neural network, a recurrent neural network, an ensemble of a plurality of neural networks, or any combination thereof, wherein the first layer of the second model includes a regression neural network, and wherein the second layer of the second model includes a logistic regression model.

9. The computing system of claim 7, wherein the at least one processor is further programmed and/or configured to:

determine a plurality of information values of a plurality of intermediate layers of the first model; and

select the intermediate layer from the plurality of intermediate layers based on the plurality of information values.

10. The computing system of claim 7, wherein the first set of features includes complex features, and wherein the second set of features includes interpretable features.

11. The computing system of claim 7, wherein the first model includes a greater number of parameters than the second model.

12. The computing system of claim 7, wherein the at least one processor is further programmed and/or configured to:

provide the trained second model;

obtain input data associated with at least one transaction; and

process, using the trained second model, the input data to generate output data, wherein the output data includes a prediction of whether the at least one transaction is a fraudulent transaction.

13. A computer program product comprising at least one non-transitory computer-readable medium including program instructions that, when executed by at least one processor, cause the at least one processor to:

obtain first training data associated with a first set of features and second training data associated with a second set of features different than the first set of features, wherein each of the first set of features and the second set of features is associated with a same plurality of transactions and a same plurality of labels for the same plurality of transactions;

train a first model based on the first training data, the second training data, and the same plurality of labels; and

train a second model, using a loss function that depends on an output of an intermediate layer of the first model and an output of the second model, based on the second training data,

wherein the second model includes at least one first layer and at least one second layer, wherein the output of the second model includes an output of the at least one first layer, and wherein the program instructions, when executed by the at least one processor, further cause the at least one processor to train the second model by:

providing, as input to the second model, only the second training data;

receiving, as output from the second model, the output of the second model;

modifying, using the loss function that depends on the output of the intermediate layer of the first model and the output of the second model including the output of the at least one first layer, one or more parameters of the at least one first layer of the second model to learn to match the output of the intermediate layer of the first model; and

training the at least one second layer based on the output of the at least one first layer and the same plurality of labels,

wherein the first model includes a recurrent neural network that includes two long short-term memory (LSTM) hidden layers including the intermediate layer of the first model, wherein the output of the intermediate layer of the first model includes a hidden state vector $h(t)$ from the second LSTM hidden layer of the two LSTM hidden layers, wherein the second model includes a feedforward regression neural network (FRNN), and wherein the loss function minimizes a squared error (L2) loss between the output of the second model and the output of the intermediate layer of the first model.

14. The computer program product of claim 13, wherein the instructions further cause the at least one processor to:

determine a plurality of information values of a plurality of intermediate layers of the first model; and

select the intermediate layer from the plurality of intermediate layers based on the plurality of information values.

15. The computer program product of claim 13, wherein the first set of features includes complex features, and wherein the second set of features includes interpretable features.

16. The computer program product of claim 13, wherein the first model includes a greater number of parameters than the second model.

17. The computer program product of claim 13, wherein the instructions further cause the at least one processor to:

provide the trained second model;

obtain input data associated with at least one transaction; and

process, using the trained second model, the input data to generate output data, wherein the output data includes a prediction of whether the at least one transaction is a fraudulent transaction.

18. The computer-implemented method of claim 1, wherein the second model includes a feedforward regression neural network FRNN, and wherein the loss function minimizes the squared error (L2) loss between the output of the second model and the output of the intermediate layer of the first model according to the following Equation:

$$\mathcal{L}(W_1, W_2) = \frac{1}{2N} \sum_{t=1}^N ||FRNN(x(t); W_1, W_2) - h(t)||^2_2$$

where $x(t)$ is a transaction of the same plurality of transactions, W_1 is a weight matrix between an input feature vector x of the transaction $x(t)$ and the at least one first layer of the second model, W_2 is a weight matrix between the at least one first layer

of the second model to the hidden state vector $h(t)$, and N is a number of the same plurality of transactions.

19. The computing system of claim 7, wherein the loss function minimizes the squared error (L2) loss between the output of the second model and the output of the intermediate layer of the first model according to the following Equation:

$$\mathcal{L}(W_1, W_2) = \frac{1}{2N} \sum_{t=1}^N ||FRNN(x(t); W_1, W_2) - h(t)||^2$$

where $x(t)$ is a transaction of the same plurality of transactions, W_1 is a weight matrix between an input feature vector x of the transaction $x(t)$ and the at least one first layer of the second model, W_2 is a weight matrix between the at least one first layer of the second model to the hidden state vector $h(t)$, and N is a number of the same plurality of transactions.

20. The computer program product of claim 13, wherein the loss function minimizes the squared error (L2) loss between the output of the second model and the output of the intermediate layer of the first model according to the following Equation:

$$\mathcal{L}(W_1, W_2) = \frac{1}{2N} \sum_{t=1}^N ||FRNN(x(t); W_1, W_2) - h(t)||^2$$

where $x(t)$ is a transaction of the same plurality of transactions, W_1 is a weight matrix between an input feature vector x of the transaction $x(t)$ and the at least one first layer of the second model, W_2 is a weight matrix between the at least one first layer of the second model to the hidden state vector $h(t)$, and N is a number of the same plurality of transactions.