Claims

1. A method for accelerating convergence of Recurrent Neural Network (RNN) for machine failure prediction, the method comprising:

setting, by a processor in a computer system, initial parameters in RNN wherein the initial parameters include an initial learning rate which is determined based on a standard deviation of a plurality of basic memory depth values identified from a machine failure sequence;

training RNN based on the initial parameters and at end of each predetermined time period,

calculating, by the processor, a current pattern error based on a vector distance between the machine failure sequence and current predicted sequence; and

if the current pattern error is less than or not greater than a predetermined error threshold value, determining, by the processor, an updated learning rate based on the current pattern error, and updating weight values between input and hidden units in RNN based on the updated learning rate.

2. The method according to claim 1, wherein the step of determining the updated learning rate based on the current pattern error comprises:

determining, by the processor, a base updated learning rate based on the current pattern error and a first predetermined rule, wherein the first

predetermined rule defines a relationship between current learning rate and the base updated learning rate based on the current pattern error.

3. The method according to claim 2, wherein the first predetermined rule comprises:

if the current pattern error is less than 0.60 and not less than 0.55, the base updated learning rate is 1/14 of the current learning rate; and

if the current pattern error is less than 0.55, the base updated learning rate is 1/2 of the current learning rate.

4. The method according to claim 2 or claim 3, wherein the step of determining the updated learning rate based on the current pattern error further comprises:

determining, by the processor, an acceleration/deceleration parameter based on whether the current pattern error is less than an immediately preceding pattern error in RNN and a second predetermined rule, wherein the second predetermined rule associates an acceleration parameter and a deceleration parameter to a predetermined pattern error range; and

calculating, by the processor, the updated learning rate based on the base updated learning rate and the acceleration/deceleration parameter.

5. The method according to claim 4, wherein the second predetermined rule comprises:

if the current pattern error is greater than or equal to 0.3, the acceleration parameter is 70%, the deceleration parameter is 30%;

if the current pattern error is less than 0.3 and greater than or equal to 0.2, the acceleration parameter is 40%, the deceleration parameter is 20%;

if the current pattern error is less than 0.2 and greater than or equal to 0.1, the acceleration parameter is 30%, the deceleration parameter is 10%; and

if the current pattern error is less than 0.1, the acceleration parameter is 10%, the deceleration parameter is 0%.

6. The method according to claim 4 or claim 5, wherein the step of determining the updated learning rate based on the current pattern error further comprises:

determining, by the processor, a bonus acceleration parameter based on the current pattern error and a third predetermined rule, wherein the third predetermined rule comprises if pattern errors have continually decreased during a predetermined count value of the predetermined time periods, a predetermined bonus acceleration parameter is to be used for calculating the updated learning rate; and

calculating, by the processor, the updated learning rate based on the base updated learning rate, the acceleration parameter and the bonus acceleration parameter if needed.

7. The method according to claim 6, wherein the predetermined count value is 10, the predetermined bonus acceleration parameter is 20% and the third predetermined rule further comprises: only if the current pattern error is not greater than 0.3, the bonus acceleration parameter is used to calculate the updated learning rate.

8. The method according to any preceding claim, further comprising:

calculating, by the processor, the initial learning rate according to the following equation:

$$\eta_0 = 2.24 - 0.08\sigma$$

wherein η_0 refers to the initial learning rate, and σ refers to the standard deviation of the basic memory depth values in the machine failure sequence.

9. The method according to any preceding claim, further comprising:

calculating, by the processor, the predetermined error threshold value according to the following equation:

$$Es = 0.03 - \sigma / 1800$$

wherein Es refers to the error threshold value, and σ refers to the standard deviation of the basic memory depth values in the machine failure sequence.

10. The method according to any preceding claim, wherein the step of setting initial parameters in RNN further comprises:

setting, by the processor, initial weights between input and hidden units in RNN as an average of weights for a plurality of elementary sequences wherein each of the plurality of elementary sequences has a basic memory depth value included in the machine failure sequence.

11. The method according to any preceding claim, further comprising:

calculating, by the processor, a cue unit by dividing a count value of time steps from the immediately preceding machine failure in the machine failure sequence by 100; and

using, by the processor, the cue unit in input layer of RNN.

12. A system for accelerating convergence of RNN for machine failure prediction, the system comprising: a computer system having a processor and a memory communicably coupled to the processor, wherein the memory is configured to store instructions to be executed by the processor, wherein the processor is configured to

set initial parameters in RNN wherein the initial parameters include an initial learning rate which is determined based on a standard deviation of a plurality of basic memory depth values identified from a machine failure sequence;

train RNN based on the initial parameters and at end of each predetermined time period;

calculate current pattern error based on a vector distance between the machine failure sequence and current predicted sequence; and

if the current pattern error is less than or not greater than a predetermined error threshold value, determine an updated learning rate based on the current pattern error, and update weight values between input and hidden units in RNN based on the updated learning rate.

13. The system according to claim 12, wherein the processor is further configured to determine a base updated learning rate based on a first predetermined rule, wherein the first predetermined rule defines a relationship between current learning rate and the base updated learning rate depending on the current pattern error.

14. The system according to claim 13, wherein the processor is further configured to determine an acceleration/deceleration parameter based on whether the current pattern error is less than an immediately preceding pattern error in RNN and a second predetermined rule, wherein the second predetermined rule associates an acceleration parameter and a deceleration parameter to a predetermined pattern error range; and

calculate the updated learning rate based on the base updated learning rate and the acceleration/deceleration parameter.

15. The system according to claim 14, wherein the processor is further configured to determine a bonus acceleration parameter based on the current

pattern error and a third predetermined rule, wherein the third predetermined rule comprises if pattern errors have continually decreased during a predetermined count value of the predetermined time periods, a predetermined bonus acceleration parameter is to be used for calculating the updated learning rate; and calculate the updated learning rate based on the base updated learning rate, the acceleration parameter and the bonus acceleration parameter.

16. The system according to any one of claim 12 to claim 15, wherein the processor is further configured to calculate the initial learning rate according to the following equation:

$$\eta_0 = 2.24 - 0.08\sigma$$

wherein $\eta_0$refers to the initial learning rate, and σ refers to the standard deviation of the basic memory depth values in the machine failure sequence.

17. The system according to any one of claim 12 to claim 16, wherein the processor is further configured to calculate the predetermined error threshold value according to the following equation:

$$Es = 0.03 - \sigma / 1800$$

wherein Es refers to the error threshold value, and σ refers to the standard deviation of the basic memory depth values in the machine failure sequence.

18. The system according to any one of claim 12 to claim 17, wherein the processor is further configured to set initial weight values between input and

hidden units in RNN as average values of weight values for a plurality of elementary sequences wherein each of the plurality of elementary sequences has a basic memory depth value included in the machine failure sequence.

19. The system according to any one of claim 12 to claim 18, wherein the processor is further configured to calculate a cue unit by dividing a count value of time steps from the immediately preceding machine failure in the machine failure sequence by 100; and use the cue unit in input layer of RNN.

20. A non-transitory computer readable medium comprising computer program code for accelerating convergence of Recurrent Neural Network (RNN) for machine failure prediction, wherein the computer program code, when executed, is configured to cause a processor in a computer system to perform a method according to any one of claim 1 to claim 11.