

CLAIMS

1. A computer-implemented method of training a neural network configured to process an input sequence and to generate a network output for the input sequence, wherein:
 - the neural network generates a respective output item for each of a plurality of input items in the input sequence, and
 - the neural network comprises a stack of layer blocks, each layer block comprising one or more neural network layers, the stack of layer blocks comprising a first layer block and a final layer block,
 - wherein the training comprises:
 - receiving an input sequence comprising a respective input item at each of a plurality of input time steps; and
 - at each of a plurality of processing time steps in a sequence of processing time steps:
 - processing the input item of an input time step corresponding to the processing time step using the first layer block to generate a first block output;
 - for each particular layer block that is not the first layer block, processing a block output generated by the preceding layer block in the stack of layer blocks at the preceding processing time step in the sequence of processing time steps using the particular layer block to generate a current block output, wherein the current block output generated by the final layer block is the output item for an input item of an earlier input time step than the input time step corresponding to the processing time step;
 - computing i) a current error in the output item generated by the final layer block at the processing time step and ii) a current gradient of the current error for the final layer block;
 - generating a parameter update for the final layer block from the current error in the output item;
 - for each particular layer block that is not the final layer block, computing a current gradient for the particular layer block from i) a preceding gradient computed by the subsequent layer block in the stack of layer blocks at the preceding processing time step in the sequence of processing time steps and ii) the preceding block output generated by the preceding layer block in the stack of layer blocks at the preceding processing time step in the sequence of processing time steps; and

for each particular layer block that is not the final layer block, generating a parameter update for the particular layer block from the preceding gradient computed by the subsequent layer block in the stack of layer blocks at the preceding processing time step in the sequence of processing time steps.

2. The method of claim 1, further comprising, at each of a plurality of second processing time steps in a sequence of second processing time steps:

processing the input item of an input time step corresponding to the second processing time step using the first layer block to generate a first block output; and

for each particular layer block that is not the first layer block, processing a block output generated by the preceding layer block in the stack of layer blocks at the preceding second processing time step in the sequence of second processing time steps using the particular layer block to generate a current block output, wherein the current block output generated by the final layer block is the output item for an input item of an earlier input time step than the input time step corresponding to the second processing time step;

computing i) a current error in the output item generated by the final layer block at the second processing time step and ii) a current gradient of the current error for the final layer block;

generating a parameter update for the final layer block from the current error in the output item; and

for each particular layer block that is not the final layer block and for which the subsequent layer block in the stack of layer blocks computed a preceding gradient at the preceding second processing time step in the sequence of second processing time steps:

computing a current gradient for the particular layer block in the stack of layer blocks from i) the preceding gradient computed by the subsequent layer block at the preceding second processing time step and ii) the current block output generated by the preceding layer block in the stack of layer blocks at the preceding second processing time step; and

generating a parameter update for the particular layer block in the stack of layer blocks from the preceding gradient computed by the subsequent layer block at the preceding second processing time step,

wherein the sequence of second processing time steps precedes the sequence of processing time steps.

3. The method of any one of claims 1 or 2, further comprising, at each of a plurality of third processing time steps in a sequence of third processing time steps:

for each particular layer block that i) generated a preceding block output at the preceding third processing time step in the sequence of third processing time steps and ii) is not the final layer block, processing the preceding block output generated by the particular layer block at the preceding third processing time step using the subsequent layer block in the stack of layer blocks to generate a current block output, wherein the current block output generated by the final layer block is the output item for an input item of an earlier input time step than the input time step corresponding to the third processing time step;

computing i) a current error in the output item generated by the final layer block at the third processing time step and ii) a current gradient of the current error for the final layer block;

generating a parameter update for the final layer block from the current error in the output item;

for each particular layer block that is not the final layer block, computing a current gradient for the particular layer block from i) a preceding gradient computed by the subsequent layer block in the stack of layer blocks at the preceding processing time step in the sequence of processing time steps and ii) the current block output generated by the particular layer block at the processing time step; and

for each particular layer block that is not the final layer block, generating a parameter update for the particular layer block from the preceding gradient computed by the subsequent layer block in the stack of layer blocks at the preceding processing time step in the sequence of processing time steps,

wherein the sequence of third processing time steps succeeds the sequence of processing time steps.

4. The method of any one of claims 1-3, further comprising, at each of a plurality of fourth processing time steps in a sequence of fourth processing time steps:

for each particular layer block that is not the final layer block and for which the subsequent layer block in the stack of layer block computed a preceding gradient at the preceding fourth processing time step in the sequence of fourth processing time steps:

computing a current gradient for the particular layer block in the stack of layer blocks from i) the preceding gradient computed by the subsequent layer block at the preceding fourth processing time step and ii) the block output most recently generated by the preceding layer block in the stack of layer blocks; and

generating a parameter update for the particular layer block in the stack of layer blocks from the preceding gradient computed by the subsequent layer block at the preceding second processing time step,

wherein the sequence of fourth processing time steps succeeds the sequence of processing time steps.

5. The method of any one of claims 1-4, wherein computing a current gradient for a particular layer block that is not the final layer block comprises:

computing a first Jacobian of the particular layer block with respect to the block output generated by the preceding layer block in the stack of layer blocks at the preceding processing time step; and

multiplying the first Jacobian with the preceding gradient computed by the subsequent layer block in the stack of layer blocks at the preceding processing time step in the sequence of processing time steps.

6. The method of any one of claims 1-5, wherein computing a current gradient for the final layer block comprises:

computing a first Jacobian of the final layer block with respect to the block output generated by the preceding layer block in the stack of layer blocks at the preceding processing time step.

7. The method of any one of claims 1-6, wherein generating a parameter update for a particular layer block that is not the final layer block comprises:

generating a second gradient for the particular layer block, comprising:

computing a second Jacobian of the particular layer block with respect to current values of the parameters of the particular layer block; and

multiplying the second Jacobian with the preceding gradient computed by the subsequent layer block in the stack of layer blocks at the preceding processing time step in the sequence of processing time steps; and

generating the parameter update from the second gradient.

8. The method of any one of claims 1-7, wherein generating a parameter update for the final layer block comprises:

generating a second gradient for the final layer block, comprising:

computing a second Jacobian of the final layer block with respect to current values of the parameters of the final layer block; and

generating the parameter update from the second gradient.

9. The method of any one of claims 1-8, wherein generating a parameter update comprises generating the parameter update using stochastic gradient descent.

10. The method of any one of claims 1-9, further comprising, for each layer block:

combining the parameter updates for the layer block generated at a plurality of respective processing time steps to generate a combined parameter update, and

updating parameters of the layer block using the combined parameter update.

11. A system comprising one or more computers and one or more storage devices storing instructions that when executed by the one or more computers cause the one more computers to perform the operations of the respective method of any one of claims 1-10.

12. One or more non-transitory computer storage media storing instructions that when executed by one or more computers cause the one more computers to perform the operations of the respective method of any one of claims 1-10.