

WHAT IS CLAIMED IS:

- 1 1. A method comprising:
2 receiving a plurality of execution traces of a program, each execution trace
3 comprising a plurality of variable values;
4 encoding, by a first recurrent neural network, the plurality of variable
5 values to generate a plurality of program states for each execution trace;
6 determining, by a bi-directional recurrent neural network, a reduced set of
7 program states for each execution trace from the plurality of program states;
8 encoding, by a second recurrent neural network, the reduced set of
9 program states to generate a plurality of executions for the program;
10 pooling the plurality of executions to generate a program embedding; and
11 predicting semantics of the program using the program embedding.
- 1 2. The method of claim 1, wherein predicting semantics of the
2 program comprises returning a probability of one of a list of program functionalities.
- 1 3. The method of claim 1, wherein there are 50% fewer states in the
2 reduced set of program states than in the plurality of program states.
- 1 4. The method of claim 1, wherein each execution trace in the plurality
2 of execution traces is generated with randomly generated inputs.
- 1 5. The method of claim 1, wherein determining a reduced set of
2 program states comprises:
3 computing a forward sequence from the plurality of program states;
4 computing a backward sequence from the plurality of program states; and
5 for each program state in the plurality of program states:
6 determining a forward context vector from the forward sequence;
7 determining a backward context vector from the backward
8 sequence; and

9 determining, based on at least the forward context vector and
10 backward context vector, to include the program state in the reduced set of program
11 states.

1 6. The method of claim 5, wherein determining the forward context
2 vector comprises pooling a subset of components of the forward sequence, the subset
3 of components of the forward sequence comprising information from a first s-1 program
4 states, and wherein determining the backward context vector comprises pooling a
5 subset of components of the backward sequence, the subset of components of the
6 backward sequence comprising information from a last m-s program states.

1 7. The method of claim 5, wherein determining to include the program
2 state in the reduced set of program states is performed by a multi-layer perceptron.

1 8. The method of claim 1, wherein predicting semantics is performed
2 with an output of a softmax function.

1 9. The method of claim 1, wherein the program has been mutated.

1 10. The method of claim 1, wherein the method is optimized to
2 minimize a cross-entropy loss and minimize the number of program states in the
3 reduced set of program states.

1 11. A computer comprising:
2 a processor; and
3 a computer-readable medium comprising code, executable by the
4 processor, for implementing a method comprising:
5 receiving a plurality of execution traces of a program, each
6 execution trace comprising a plurality of variable values;
7 encoding, by a first recurrent neural network, the plurality of
8 variable values to generate a plurality of program states for each execution trace;
9 determining, by a bi-directional recurrent neural network, a reduced
10 set of program states for each execution trace from the plurality of program
11 states;

12 encoding, by a second recurrent neural network, the reduced set of
13 program states to generate a plurality of executions for the program;
14 pooling the plurality of executions to generate a program
15 embedding; and
16 predicting semantics of the program using the program embedding.

1 12. The computer of claim 11, wherein predicting semantics of the
2 program comprises returning a probability of one of a list of program functionalities.

1 13. The computer of claim 11, wherein there are 50% fewer states in
2 the reduced set of program states than in the plurality of program states.

1 14. The computer of claim 11, wherein each execution trace in the
2 plurality of execution traces is generated with randomly generated inputs.

1 15. The computer of claim 11, wherein determining a reduced set of
2 program states comprises:
3 computing a forward sequence from the plurality of program states;
4 computing a backward sequence from the plurality of program states; and
5 for each program state in the plurality of program states:
6 determining a forward context vector from the forward sequence;
7 determining a backward context vector from the backward
8 sequence; and
9 determining, based on at least the forward context vector and
10 backward context vector, to include the program state in the reduced set of program
11 states.

1 16. The computer of claim 15, wherein determining the forward context
2 vector comprises pooling a subset of components of the forward sequence, the subset
3 of components of the forward sequence comprising information from a first s-1 program
4 states, and wherein determining the backward context vector comprises pooling a
5 subset of components of the backward sequence, the subset of components of the
6 backward sequence comprising information from a last m-s program states.

- 1 17. The computer of claim 15, wherein determining to include the
2 program state in the reduced set of program states is performed by a multi-layer
3 perceptron.
- 1 18. The computer of claim 11, wherein predicting semantics is
2 performed with an output of a softmax function.
- 1 19. The computer of claim 11, wherein the program has been mutated.
- 1 20. The computer of claim 11, wherein the computer is optimized to
2 minimize a cross-entropy loss and minimize the number of program states in the
3 reduced set of program states