

WHAT IS CLAIMED IS:

1. A method comprising:

receiving a plurality of execution traces of a program, each execution trace comprising a plurality of variable values, and wherein each execution trace of the program includes a semantic label describing a processing function of the program;

encoding, by a first recurrent neural network, the plurality of variable values to generate a plurality of program states for each execution trace;

determining, by a bi-directional recurrent neural network, a reduced set of program states for each execution trace from the plurality of program states, wherein determining the reduced set of program states for each execution trace from the plurality of program states comprises:

computing a forward sequence from the plurality of program states;

computing a backward sequence from the plurality of program states;

for each program state in the plurality of program states:

determining a forward context vector from the forward sequence;

determining a backward context vector from the backward sequence; and

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

encoding, by a second recurrent neural network, the reduced set of program states to generate a plurality of executions for the program;

pooling the plurality of executions to generate a program embedding; and

predicting semantics of the program using the program embedding in order to classify the program, wherein the predicted semantics include one or more semantic labels.

2. The method of claim 1, wherein the predicting semantics of the 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~~

~~_____determining, based on at least the forward context vector and backward context vector, to include the program state in the reduced set of program states.~~

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

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

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

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

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

110. A computer comprising:
a processor; and
a computer-readable medium comprising code, executable by the processor, for implementing a method comprising:
receiving a plurality of execution traces of a program, each execution trace comprising a plurality of variable values, and wherein each execution trace of the program includes a semantic label describing a processing function of the program;
encoding, by a first recurrent neural network, the plurality of variable values to generate a plurality of program states for each execution trace;

11 determining, by a bi-directional recurrent neural network, a reduced
12 set of program states for each execution trace from the plurality of program
13 states;

14 wherein determining the reduced set of program states for each
15 execution trace from the plurality of program states comprises:

16 computing a forward sequence from the plurality of program states;

17 computing a backward sequence from the plurality of program
18 states;

19 for each program state in the plurality of program states:

20 determining a forward context vector from the forward
21 sequence;

22 determining a backward context vector from the backward
23 sequence; and

24 determining, based on at least the forward context vector
25 and backward context vector, to include the program state in the reduced set of
26 program states;

27 encoding, by a second recurrent neural network, the reduced set of
28 program states to generate a plurality of executions for the program;
29 pooling the plurality of executions to generate a program
30 embedding; and
31 predicting semantics of the program using the program embedding
32 in order to classify the program, wherein the predicted semantics include one or
33 more semantic labels.

1 ~~12~~11. The computer of claim ~~11~~ 10, wherein the predicting semantics of
2 the program comprises returning a probability of one of a list of program functionalities.

1 ~~13~~12. The computer of claim ~~11~~ 10, wherein there are 50% fewer states
2 in the reduced set of program states than in the plurality of program states.

1 ~~14~~13. The computer of claim ~~11~~ 10, 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~~14. The computer of claim ~~15~~ 10, wherein determining the forward
2 context vector comprises pooling a subset of components of the forward sequence, the
3 subset of components of the forward sequence comprising information from a first s-1
4 program states, and wherein determining the backward context vector comprises

- 5 pooling a subset of components of the backward sequence, the subset of components
- 6 of the backward sequence comprising information from a last m-s program states.

1 ~~17~~15. The computer of claim ~~15~~ 10, 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~~16. The computer of claim ~~11~~ 10, wherein the predicting semantics is
2 performed with an output of a softmax function.

1 ~~19~~17. The computer of claim ~~11~~ 10, wherein the program has been
2 mutated.

1 ~~20~~18. The computer of claim ~~11~~ 10, wherein the computer is optimized to
2 minimize a cross-entropy loss and minimize ~~the a~~ a number of program states in the
3 reduced set of program states.