

Preliminary Results

Kieran Groble, Remy Bubulka, James Gibson

October 27, 2017

Data

Our data is generated from a Prolog program. We use a given maximum depth and a small set of operations to produce all possible programs which do not produce unbound variable errors. Each program is stored as one line in a text file. We then give this file as input to an interpreter, which evaluates each file and produces the output of the program and the type of error. This can be processed by Python and fed into a Tensorflow or Keras network.

Network

Our best results so far is from a neural network with a 256 LSTM layer, a 0.2 dropout layer, and 128-64-16-1 dense layers with RELU activation functions. This was trained with 100 epochs (stopped at 58 with early stopping) with a 100 batch size of 500.

Results

Our final accuracy was 0.776 using output prediction and our error prediction accuracy was 0.969. Below are several sample outputs.

Input

```
tests = tokenize(np.array(['(+ 1 1)',  
                           '((lambda) (x) (+ x 1)) 1)',  
                           '(/ 1 2)',  
                           '(* 2 (* 2 2))',  
                           '(* 2 2)',  
                           '(* 1 (+ 2 1))',
```

Output

```
array([[ 2.18065166],  
       [ 1.73664606],  
       [ 0.54599768],  
       [ 7.64976978],  
       [ 3.83102441],  
       [ 2.84936213],  
       [ 3.4980514 ]], dtype=float32)
```