

Teaching the Art of Computer Programming at a Distance by Generating Dialogues using Deep Neural Networks

Yijun Yu¹, Anton Dil¹, Irum Rauf¹

¹The Open University, UK

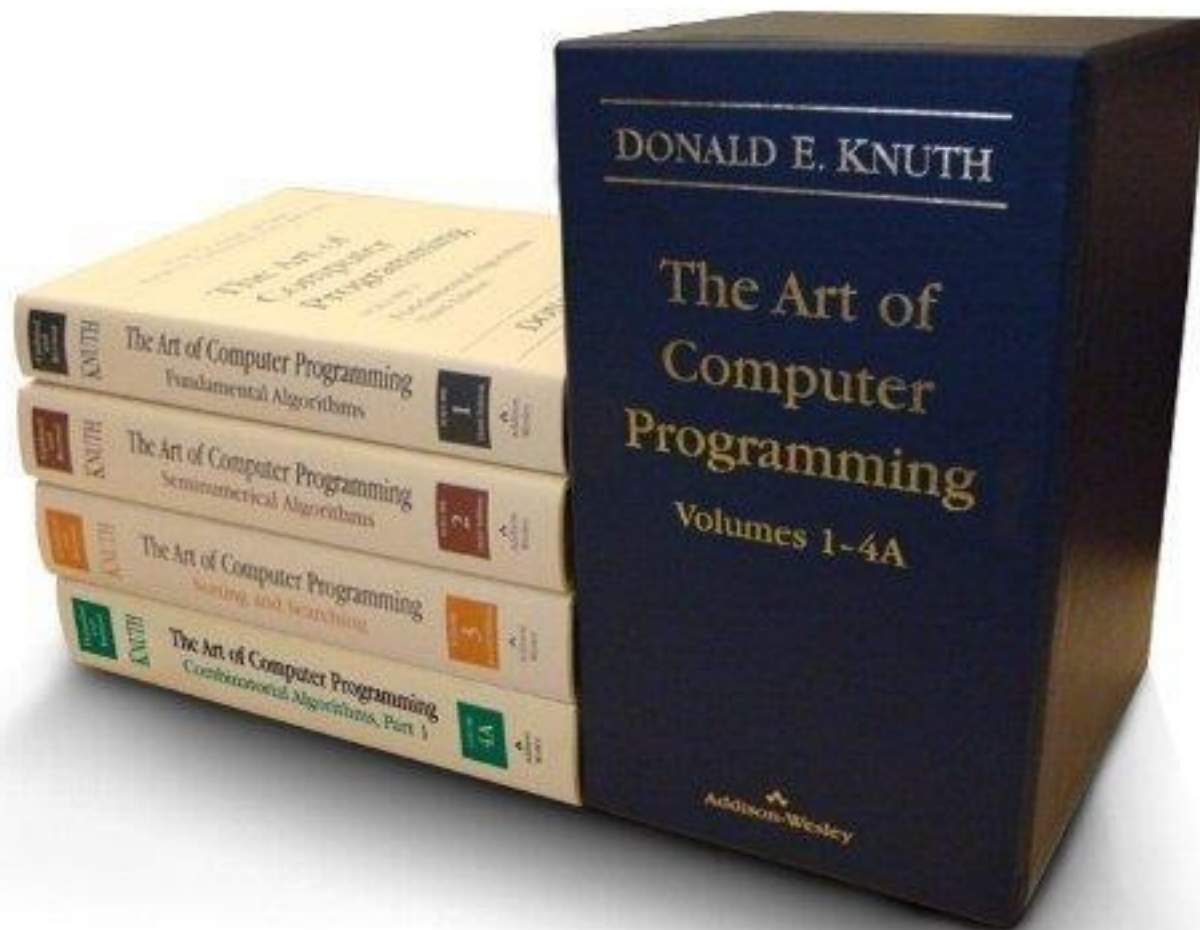
Xiaozhu Wang²

²The Open University of China



*November 6, 2019 @ Convention Centre Dublin, Ireland
28th ICDE - World Conference on Online Learning*

The Art of Computer Programming



Finding an error in a Knuth text.

Stupidity: Cashing that \$2.56 check you got.

DONALD E. KNUTH
COMPUTER SCIENCE DEPARTMENT
STANFORD UNIVERSITY
STANFORD, CA 94305-9045

11-3167/1210
01

505


Date 8 Mar 99

Pay to the
Order of

\$ 2.56

Two and 56/100

Dollars

 Security features
are included.
Details on back.

AMERICA CALIFORNIA BANK

2390 El Camino Real • Palo Alto, CA 94306

For 8.589

Donald Knuth MP

⑆ 1291 31673⑆ 0505 01 14584906 ⑈

Algorithms

Program S (*Straight insertion sort*). The records to be sorted are in locations INPUT+1 through INPUT+N; they are sorted in place in the same area, on a full-word key. $rI1 \equiv j - N$; $rI2 \equiv i$; $rA \equiv R \equiv K$; assume that $N \geq 2$.

01	START	ENT1	2-N	1	<u>S1. Loop on j. $j \leftarrow 2$.</u>
02	2H	LDA	INPUT+N,1	$N - 1$	<u>S2. Set up i, K, R.</u>
03		ENT2	N-1,1	$N - 1$	$i \leftarrow j - 1$.
04	3H	CMPA	INPUT,2	$B + N - 1 - A$	<u>S3. Compare $K : K_i$.</u>
05		JGE	5F	$B + N - 1 - A$	To S5 if $K \geq K_i$.
06	4H	LDX	INPUT,2	B	<u>S4. Move R_i, decrease i.</u>
07		STX	INPUT+1,2	B	$R_{i+1} \leftarrow R_i$.
08		DEC2	1	B	$i \leftarrow i - 1$.
09		J2P	3B	B	To S3 if $i > 0$.
10	5H	STA	INPUT+1,2	$N - 1$	<u>S5. R into R_{i+1}.</u>
11		INC1	1	$N - 1$	
12		J1NP	2B	$N - 1$	$2 \leq j \leq N$. █

Teaching Java at a distance

Pedagogy of The Open University module M250

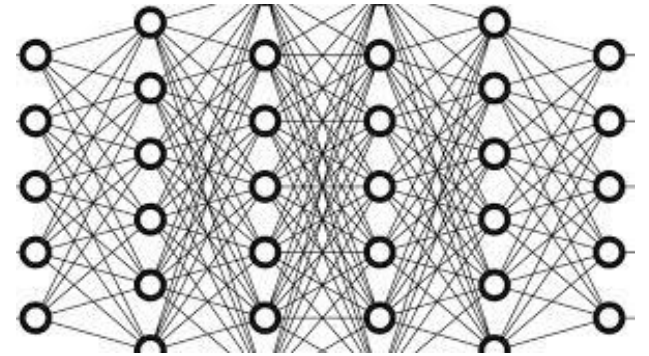
- 2nd year flagship undergraduate module at the School of Computing and Communications
- **1400** students per presentation
- Unique challenges and traditional distance teaching approaches
 - Recruit at least 1 associated lecturer per 20 students
 - Tutoring is mostly done through online materials, but also through face to face consultations
 - Tutor Marked Assignments (TMAs) and close book Exams
 - Interactive Computer Marked Assignments (iCMA)

An algorithm in Java

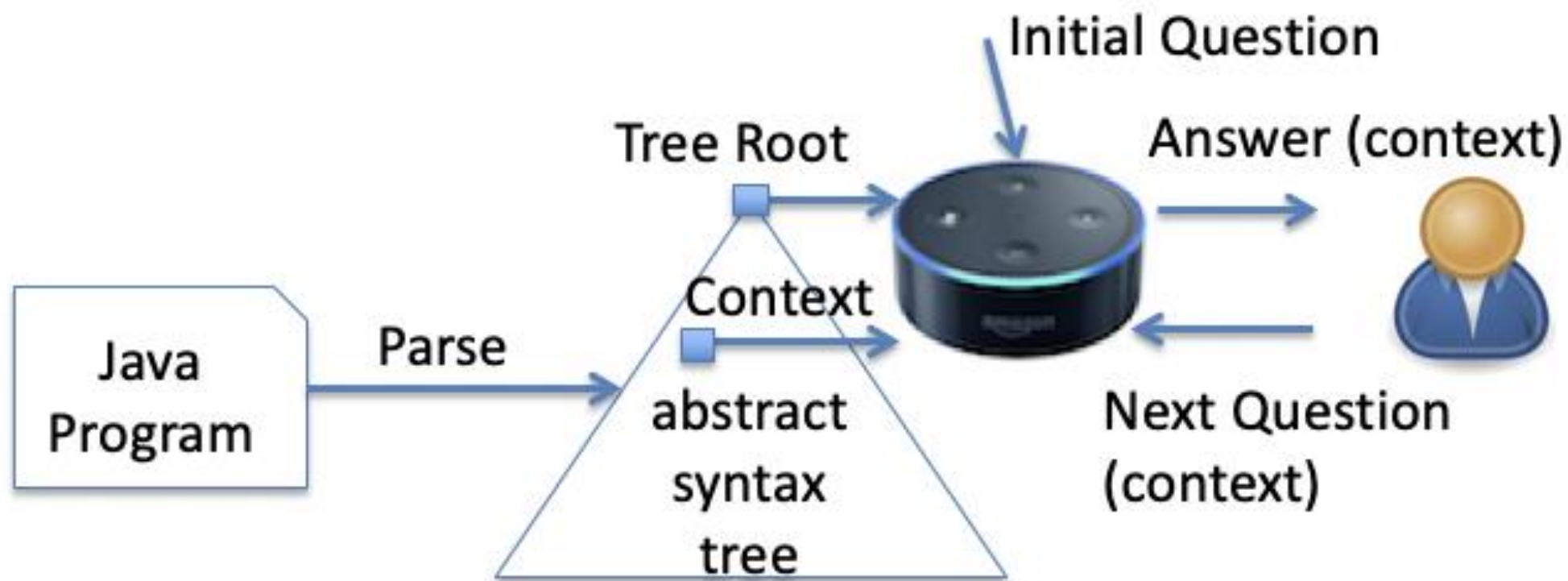
```
1  public static void insertionSort()
2
3  {
4  int i = 1;
5
6  while
7
8      (i < arr.length)
9  {
10     int j = i;
11     while
12
13         (j > 0)
14     {
15         if (arr[j].key < arr[j-1].key)
16         {
17             InsertionSortNode tmp = arr[j];
18             arr[j] = arr[j-1];
19             arr[j-1] = tmp;
20         }
21         j = j - 1;
22     }
23     i = i + 1;
24 }
25 }
26
27
```

The needs for AI

- Neural Networks
 - Image classification (convolutional)
 - Recurrent NN
 - Reinforced Learning
 - ...
- Unique challenges
 - Unlike natural languages, algorithms are structured
 - Algorithms are agnostic to programming languages
 - They are more germane to the machine, i.e., assembler
 - Students still need to learn in a natural way



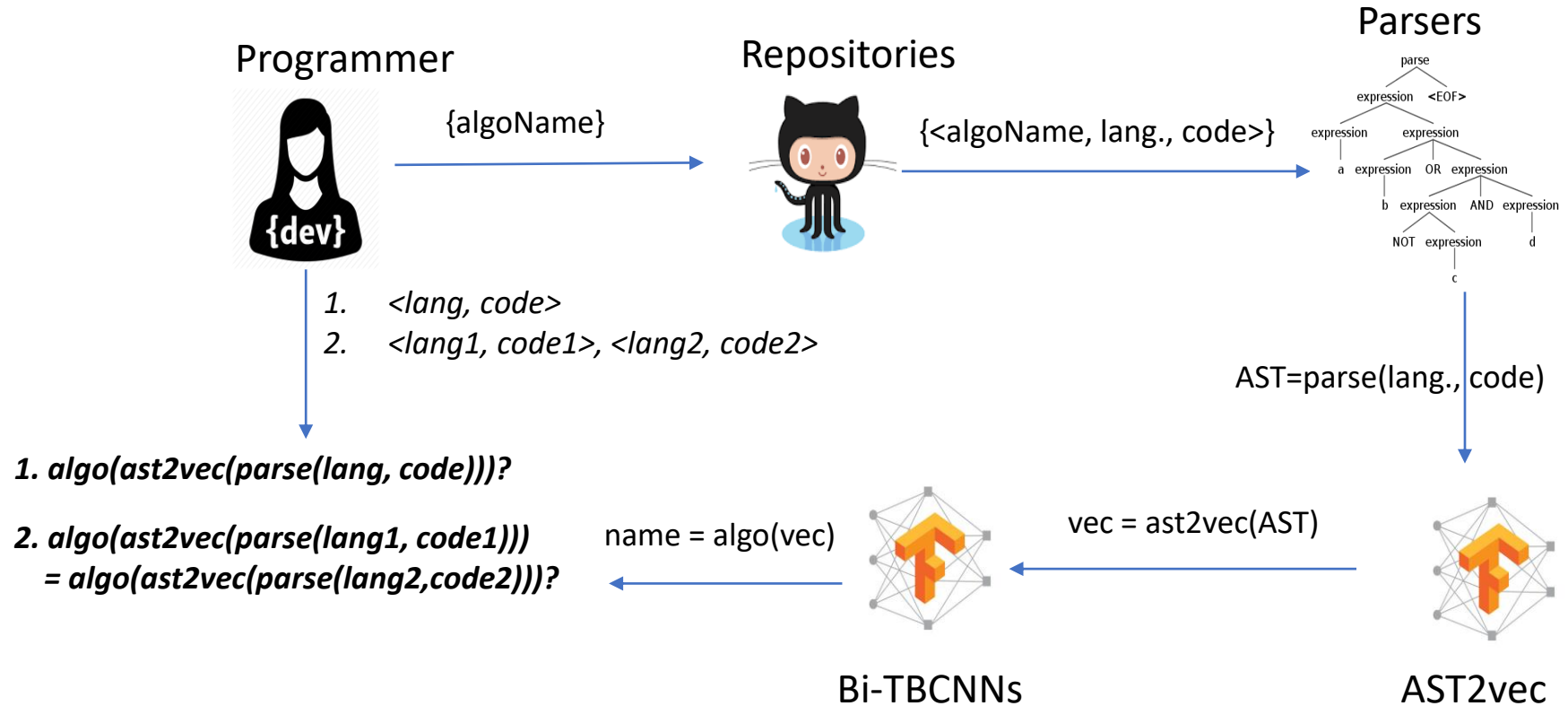
Overview: Dialogues with Echo



Knowledge Representation for Code

- What are the key features of an algorithm
 - Abstract Syntax Trees
 - Dependence Graphs
 - Def-Use chains
- What are not key features
 - Exact names of variables/methods
- Suitable representations
 - Tree-based Convolution Neural Networks [AAAI'16]
 - Gated Graph Neural Networks [SANER'19]
 - Tree Capsules Networks [FSE'19]

The Machine Learning Process



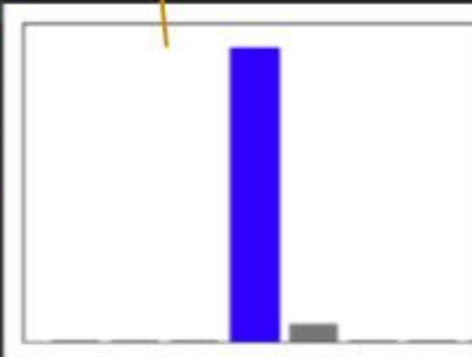
<https://github.com/yijunyu/bi-tbcnn>

<https://github.com/yijunyu/demo>

Algorithm Classification
Probability of Correct Prediction

Source code editor
(insertion sort)

Visual
Explanation
Of Classification
Results



```
1 public static void insertionSort()
2
3 {
4     int i = 1;
5
6     while
7
8         (i < arr.length)
9     {
10         int j = i;
11         while
12
13             (j > 0)
14         {
15             if (arr[j].key < arr[j-1].key)
16             {
17                 InsertionSortNode tmp = arr[j];
18                 arr[j] = arr[j-1];
19                 arr[j-1] = tmp;
20             }
21             j = j - 1;
22         }
23         i = i + 1;
24     }
25 }
26
27
```

```
public static void insertionSort()
{
    int i = 1;
    while
    {
        (i < arr.length)
        {
            int j = i;
            while
            {
                (j > 0)
                {
                    if (arr[j].key < arr[j-1].key)
                    {
                        InsertionSortNode tmp = arr[j];
                        arr[j] = arr[j-1];
                        arr[j-1] = tmp;
                    }
                    j = j - 1;
                }
            }
            i = i + 1;
        }
    }
}
```

Classification probabilities (softmax values)

2.568557159975171089e-03
4.263356095179915428e-04

Pick a Program

Problems Open Ports /workspace/demo x /workspace/demo
WARNING:tensorflow:From /workspace/.pip-modules/lib/python3.7/site-packages/tensorflow/python/training/saver.py:1266: checkpoint_exists (from tensorflow.pyt
hon.training.checkpoint_management) is deprecated and will be removed in a future version.
Instructions for updating:
Use standard file APIs to check for files with this prefix.
gitpod /workspace/demo \$

Conclusions and Future work

- Summary
 - With student's *unstructured* inputs and the underlying **structured** knowledge representations, NNs are effective in generating dialogues with *unstructured* outputs.
 - Dialogues can take different forms:
 - An Interactive Visual Programming Environment [ICSE'19]
 - Structured Arguments [JSS'15]
 - Chatbots [EPSRC Johnny project]
- Future work
 - iCMA replacement with smarter feedback (M250)
 - Evaluation with the examinable components, i.e. automated marking, using Generative Adversarial Networks [FSE'19]

References

- A Dil, Y Yu, et al. “Object-Oriented Programming in Java”, <http://www.open.ac.uk/courses/modules/m250>
- NDQ Bui, Y Yu, L Jiang. “SAR: Learning Cross-Language API Mappings with Little Knowledge”. In ESEC/FSE’19. pp. 796-806. https://github.com/bdqngchi/SAR_API_mapping
- NDQ Bui, Y Yu, L Jiang. “Bilateral Dependency Neural Networks for Cross-Language Algorithm Classification”, In SANER’19. pp. 422-433. <https://github.com/bdqngchi/bi-tbcnn>
- NDQ Bui, L Jiang, Y Yu. “Cross-language learning for program classification using bilateral tree-based convolutional neural networks”, In AAI’18. pp. 758-761. <https://github.com/bdqngchi/ggnn.tensorflow>
- Y Yu. “fAST: flattening abstract syntax trees for efficiency”, In ICSE’19, pp. 278-279. <https://github.com/yijunyu/demo>
- Y Yu, VNL Franqueira, TT Tun, RJ Wieringa, and B Nuseibeh (2015). “Automated analysis of security requirements through risk-based argumentation”. Journal of Systems and Software, 106:102–116. <https://sead.open.ac.uk/openargue>
- I Rauf, M Petre, B Nuseibeh, A Rashid. “Why Johnny doesn’t write secure software? Secure Software Development by the Masses”, <https://www.writingsecuresoftware.org>