Recurrent Inference in Text Editing

Ning Shi Ziheng Zeng Haotian Zhang Yichen Gong

ning.shi@gatech.edu zzeng13@illinois.edu haotian.zhang@learnable.ai yichen01.gong@horizon.ai

Georgia Tech UIUC Learnable.ai Horizon Robotics

Introduction

- Many text editing tasks can be solved by multiple independent inference steps recurrently
- A novel recurrent inference process, Recurrence, to tear the task into iterations of editing actions
- Three text editing tasks
  - Arithmetic Operators Restoration (AOR)
  - Arithmetic Equation Simplification (AES)
  - Arithmetic Equation Correction (AEC)
- Improvements over End2end and Tagging

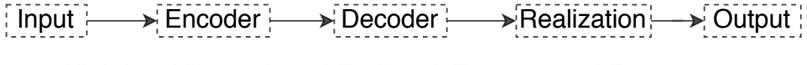
Data and Code

https://github.com/ShiningLab/Recurrent-Text-Editing

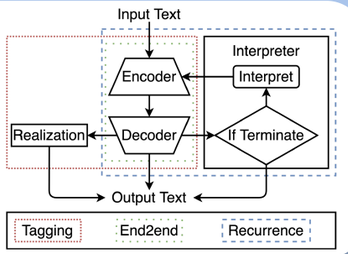
Baseline (Linear Sequential Flow)

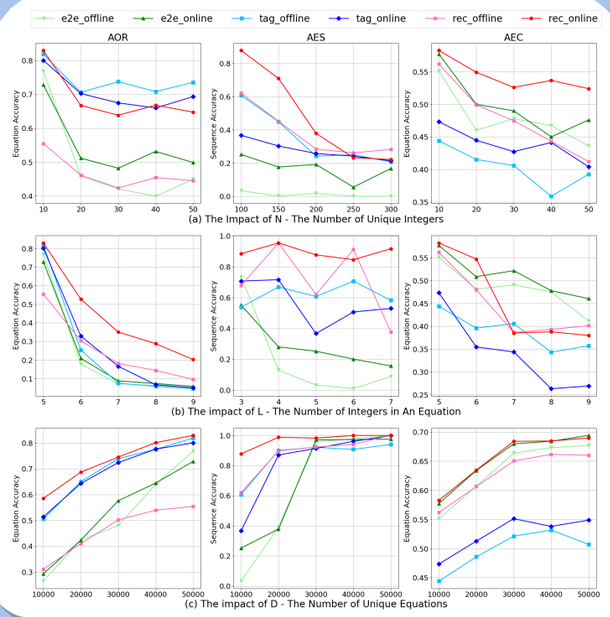
End2end

Tagging

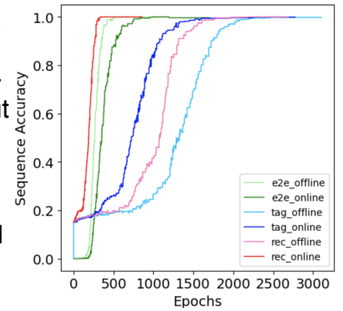


High-level illustration of End2end, Tagging, and Recurrence.





For easy settings, all three methods can achieve near-perfect results, but $\text{Recurrence}_{\text{Online}}$ converges the fastest (AES with N = 10, L = 5, and D = 10K.).

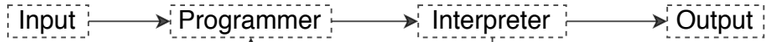


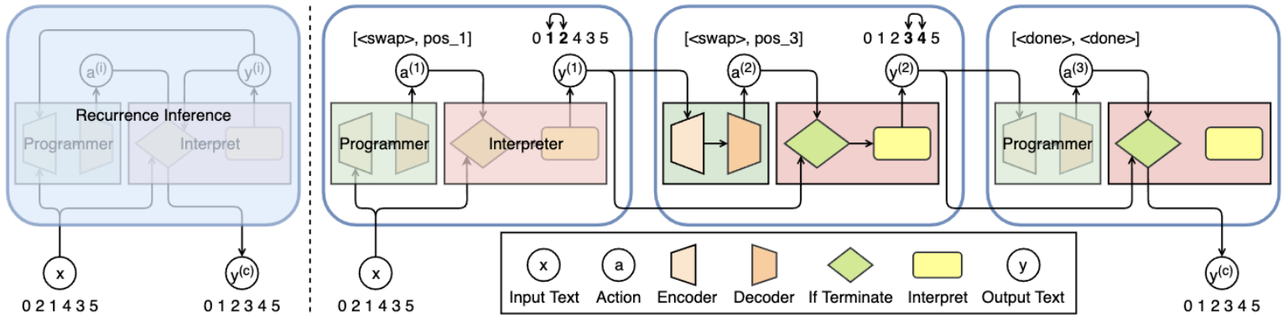
Analysis

- N - the number of unique integers (control the vocabulary size)
- L - the number of integers in an equation (control the sequence length)
- D - the number of unique equations (control the data size)

Recurrence

- Programmer, a seq2seq model to determine the editing action given the text state
- Editing Action - [Operation, Position, Symbol]
- Interpreter, a parameter-free function to excute the editing action





The number sequence [0; 2; 1; 4; 3; 5] is edited to [0; 1; 2; 3; 4; 5] via action a(1), [<swap>; pos 1], action a(2), [<swap>; pos 3], and action a(3) = [<done>;<done>], like bubble sort.

Algorithm 1: Recurrence

Result: $\mathbf{y}^{(\text{complete})}$

$\mathbf{x}_{\text{Input}} = \mathbf{x}$;
Terminate = False;
$t = 0$;
while *Terminate is not True* **do**
  $t = t + 1$;
  $\mathbf{a}^{(t)} = \texttt{Programmer}(\mathbf{x}_{\text{Input}})$;
  $\mathbf{y}^{(t)}$, Terminate $= \texttt{Interpreter}(\mathbf{x}_{\text{Input}}, \mathbf{a}^{(t)})$;
  $\mathbf{x}_{\text{Input}} = \mathbf{y}^{(t)}$;
end
$\mathbf{y}^{(\text{complete})} = \mathbf{y}^{(t)}$;

Data & Results

	AOR ($N = 10, L = 5$)	**AES** ($N = 100, L = 5$)	**AEC** ($N = 10, L = 5$)
Source	8 2 8 4 2	$-33+25+75-60 == (30-23)$	$7*8/4\ 8\ 2 - == 6$
$\text{Target}_{\text{End2end}}$	$-8*2/8+4 == 2$	$-33+25+75-60 == 7$	$7*8/4-8 == 6$
$\text{Target}_{\text{Tagging}}$	<insert_−><keep><insert_*><keep> <insert_/><keep><insert_+><keep> <insert_==><keep>	<keep><keep><keep><keep><keep> <keep><keep><keep><keep><sub_7> <delete><delete><delete><delete>	<keep><keep><keep><keep> <keep><delete><sub_−><sub_8> <keep><keep>
$\text{Target}_{\text{Recurrence}}$	<pos_0> −	<pos_9><pos_13> 7	<delete><pos_5><pos_5>

		AOR ($N = 10, L = 5, D = 10\text{K}$)		**AES** ($N = 100, L = 5, D = 10\text{K}$)			**AEC** ($N = 10, L = 5, D = 10\text{K}$)			
Method	**Training**	**#Epoch**	**Equ Acc.%**	**#Epoch**	**Token Acc.%**	**Seq Acc.%**	**#Epoch**	**Token Acc.%**	**Seq Acc.%**	**Equ Acc.%**
End2end	Offline	3352	26.47	5063	75.49	3.27	72144	87.78	54.67	55.13
	Online	2640	29.33	7795	84.60	25.20	112482	**88.08***	57.27	57.73
Tagging	Offline	1149	50.53	5223	90.10	43.80	135729	82.29	44.20	44.40
	Online	2245	51.40	4520	87.00	36.67	112968	84.46	46.93	47.33
Recurrence	Offline	1281	31.13	7603	94.92	62.07	203067	81.85	55.87	56.20
	Online	1898	**58.53***	7088	**98.63***	**87.73***	152982	83.64	**57.47***	**58.27***

Offline Training - pairs of source text and target (text, tags, and immediate editing actions)
Online Training - pairs of source and target, as well as all the intermediate text states (editing actions)