Learning by Fixing: Solving Math Word Problems with Weak Supervision



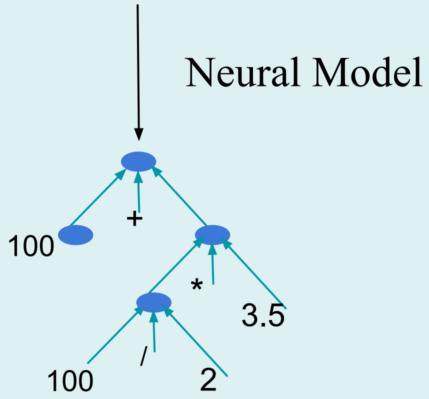


Yining Hong[1] Qing Li[2] Daniel Ciao[1] Siyuan Huang[2] Song-Chun Zhu[1,2]

[1] Department of Computer Science, UCLA [2] Department of Statistics, UCLA

Solving Math Word Problems via Neural-Symbolic Model

Problem: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?



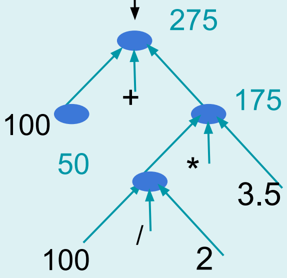
Expression Tree

Solving Math Word Problems via Neural-Symbolic Model

Problem: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?

Neural Model

Expression Tree



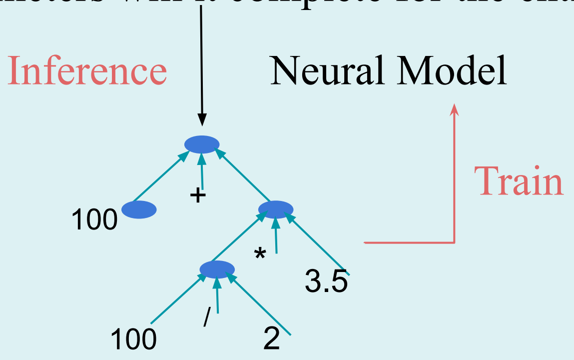
Symbolic Execution

Answer: 275

Solving Math Word Problems with Full Supervision

Problem: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?

Expression Tree (Annotated):

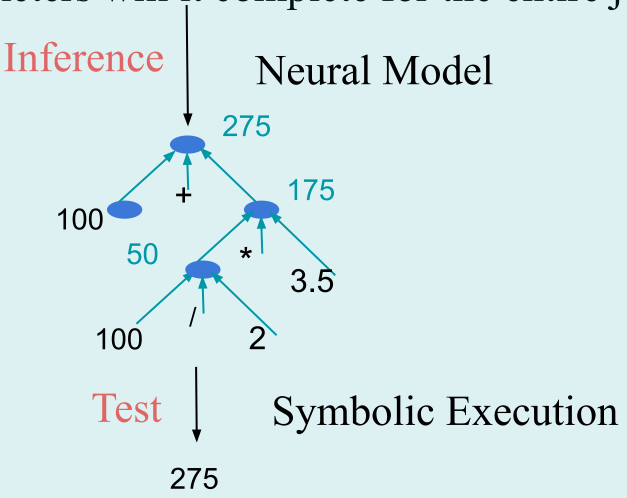


Train: only go through the neural module, optimize expression accuracy

Solving Math Word Problems with Full Supervision

Problem: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?

Expression Tree :



Answer:

Test: go through the neural module and symbolic module, evaluate the answer accuracy

Solving Math Word Problems with Full Supervision

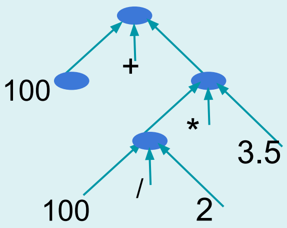
Problem: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?

Inference

Neural Model

Train

Expression Tree :



Test

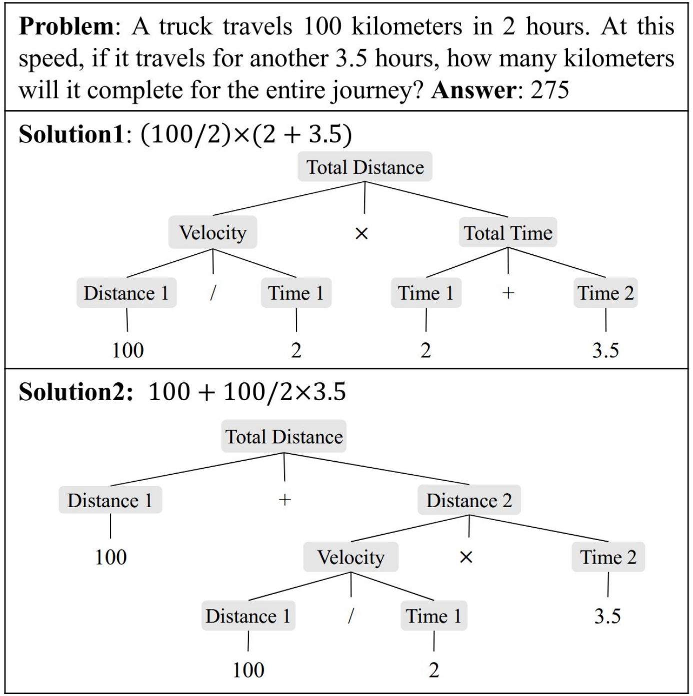
Symbolic Execution

Answer: 275

Train: only go through the neural module, optimize expression accuracy
Test: go through the neural module and symbolic module, evaluate the answer accuracy

Discrepancy

Multiple Solutions for a given math word problem



Fully-Supervised methods: fit the given solution and cannot generate diverse solutions.

Fully-supervised methods: Need time-consuming annotations

Annotating the expressions for MWPs is time-consuming.However, a large amount of MWPs with their final answers can be mined effortlessly from the internet (e.g., online forums). How to efficiently utilize these partially-labeled data without the supervision of expressions remains an open problem.

Solving Math Word Problems with Weak Supervision

Problem: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?

Expression Tree

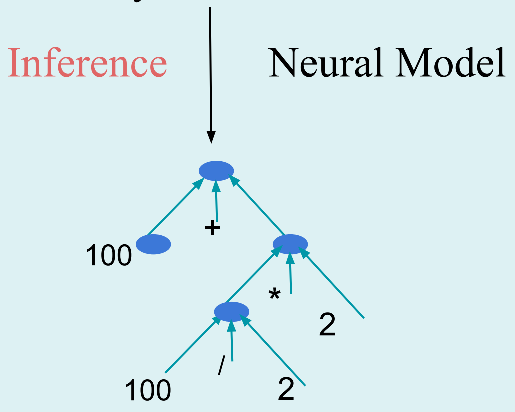
(Unannotated)

Answer (Annotated): 275

Solving Math Word Problems with Weak Supervision

Problem: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?

Expression Tree (Unannotated):



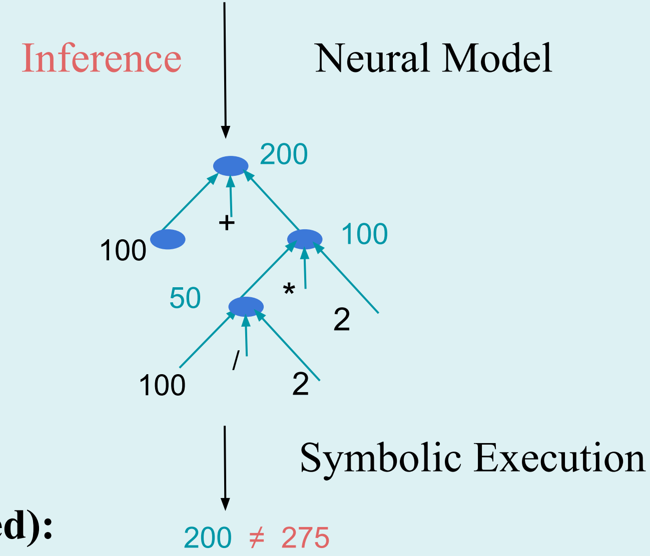
Answer (Annotated): 275

Solving Math Word Problems with Weakly Supervision

Problem: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?

Expression Tree

(Unannotated):



Answer (Annotated):

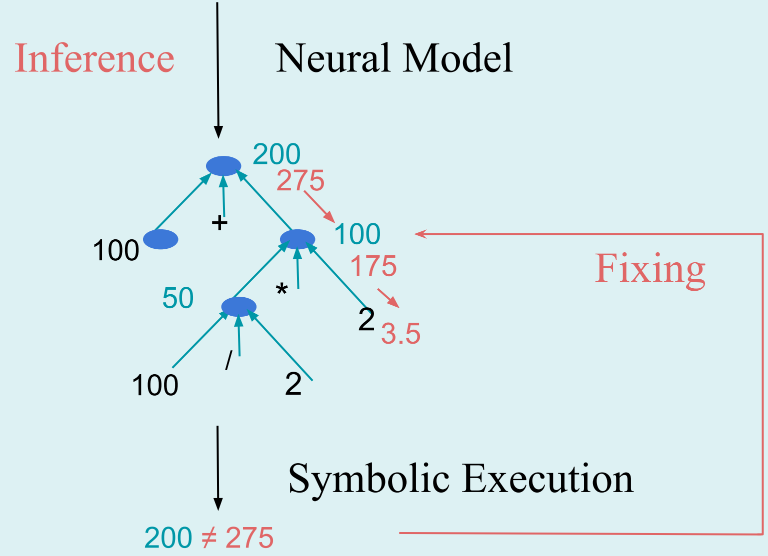
Solving Math Word Problems with Weakly Supervision

Problem: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?

Expression Tree

(Unannotated):

Answer:



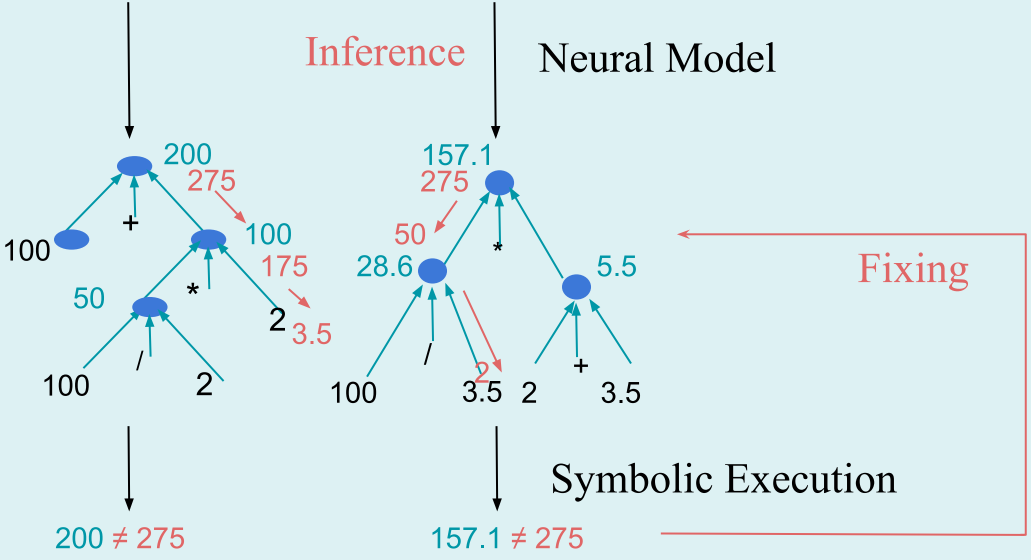
Solving Math Word Problems with Weakly Supervision

Problem: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?

Expression Tree

(Unannotated):

Answer:

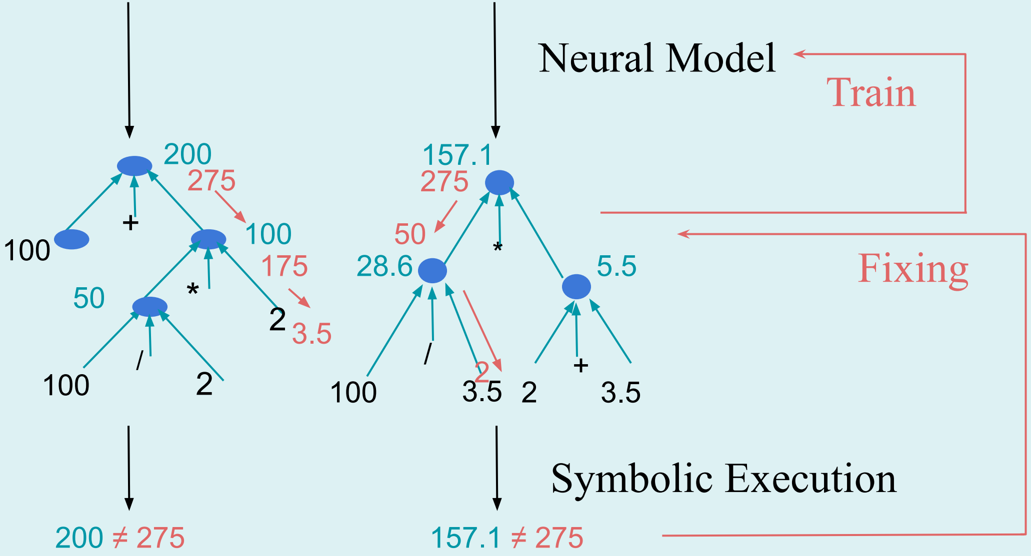


Solving Math Word Problems with Weakly Supervision

Problem: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?

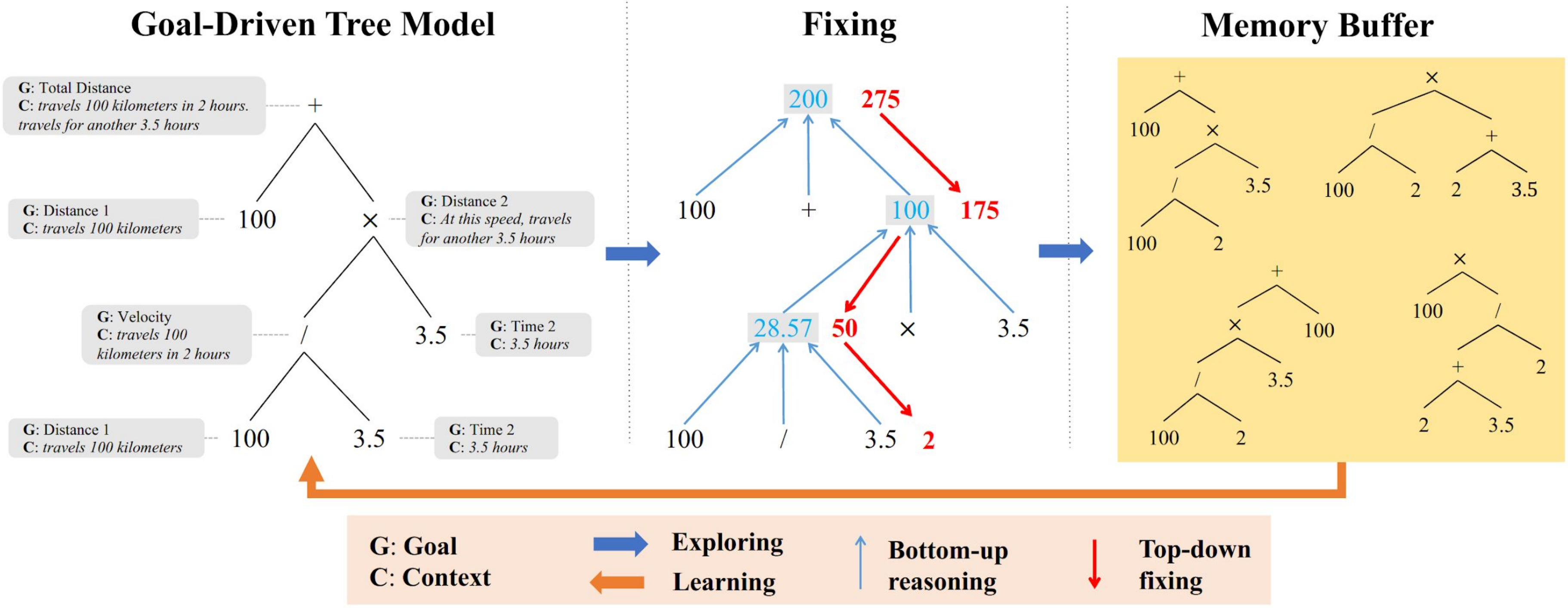
Expression Tree

(Unannotated):



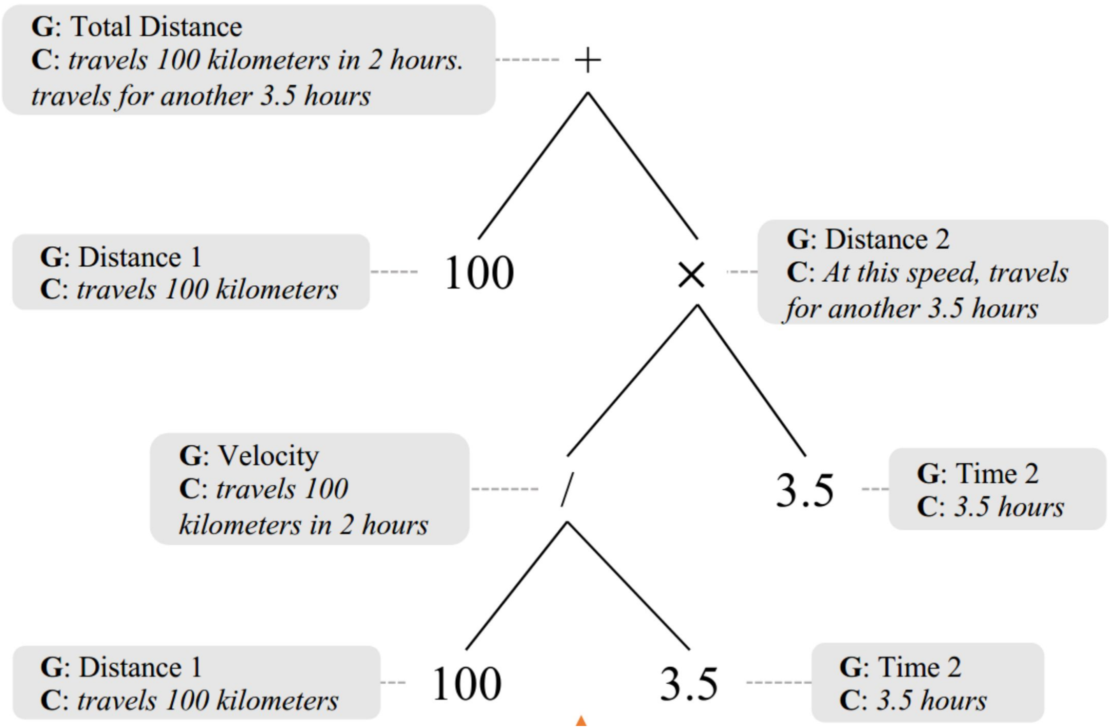
Answer:

Framework



Goal-Driven Tree Model
Fixing
Memory Buffer
G: Total Distance
C: travels 100 kilometers in 2 hours. travels for another 3.5 hours
G: Distance 1
C: travels 100 kilometers
G: Distance 2
C: At this speed, travels for another 3.5 hours
G: Velocity
C: travels 100 kilometers in 2 hours
G: Time 2
C: 3.5 hours
G: Distance 1
C: travels 100 kilometers
G: Time 2
C: 3.5 hours
200
275
100
175
28.57
50
2
G: Goal
C: Context
Exploring
Learning
Bottom-up reasoning
Top-down fixing

Goal-Driven Tree Structured Model[1]

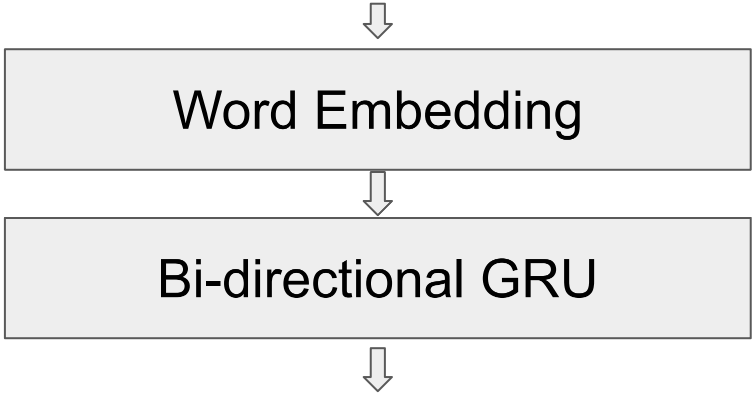


[1] A Goal-Driven Tree-Structured Neural Model for Math Word Problems. Zhipeng Xie and Shichao Sun.

Goal-Driven Tree Structured Model

- Word embedding + bi-directional GRU

Problem: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?



$$\mathbf{q}_0 = \overrightarrow{\mathbf{h}}_n^x + \overleftarrow{\mathbf{h}}_0^x$$

Goal-Driven Tree Structured Model

P: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?

$$\mathbf{W} \in R^{n \times d}$$

$$\mathbf{q}_0 = \overrightarrow{\mathbf{h}}_n^x + \overleftarrow{\mathbf{h}}_0^x$$

Attention

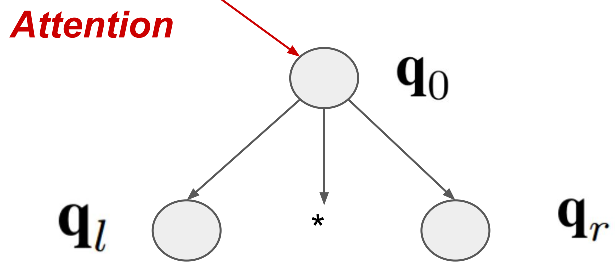
$\mathbf{q}_0$

Goal-Driven Tree Structured Model

P: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, <u>how many kilometers will it complete for the entire journey</u>?

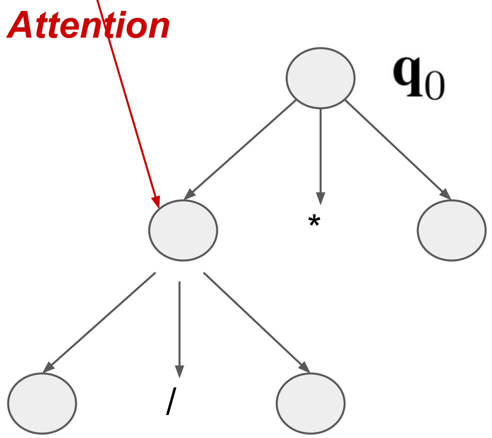
$$\mathbf{W} \in R^{n \times d}$$

$$\mathbf{q}_0 = \overrightarrow{\mathbf{h}}^x_n + \overleftarrow{\mathbf{h}}^x_0$$



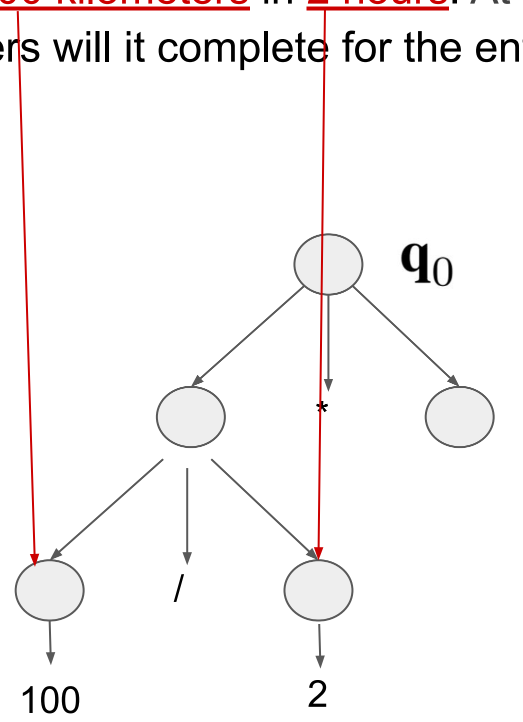
Goal-Driven Tree Structured Model

P: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?



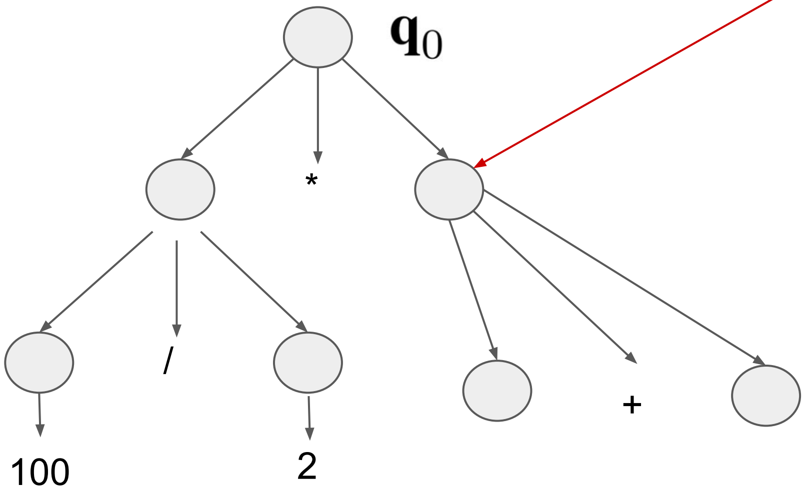
Goal-Driven Tree Structured Model

P: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?



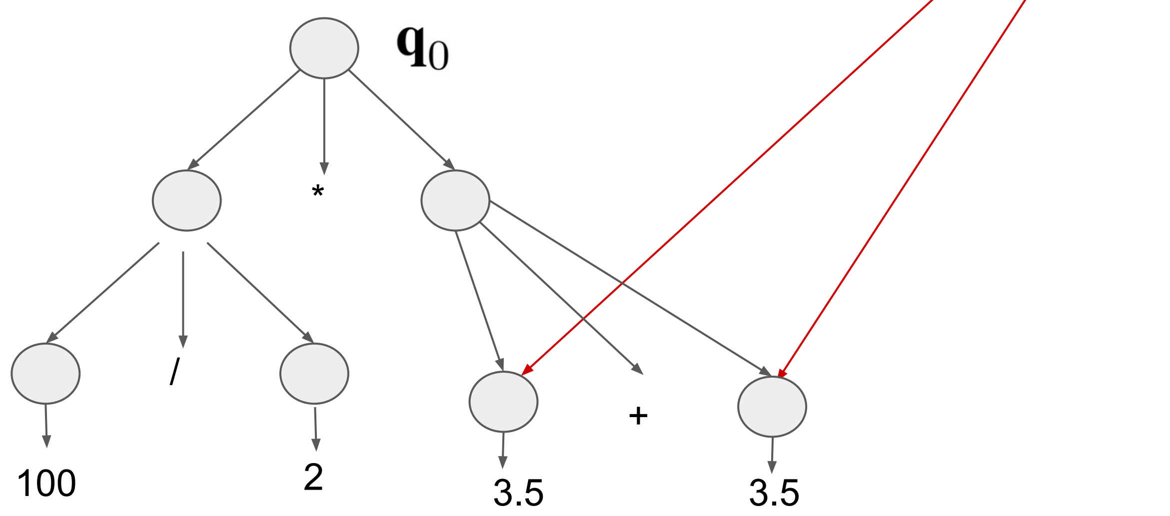
Goal-Driven Tree Structured Model

P: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?



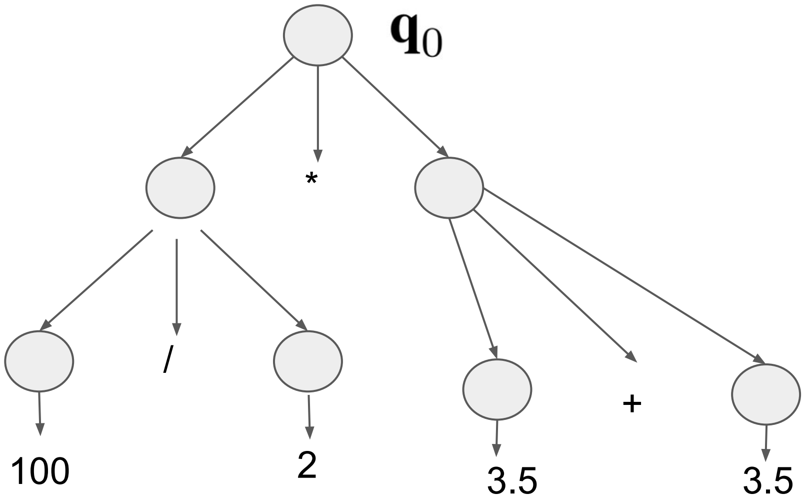
Goal-Driven Tree Structured Model

P: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for <u>another 3.5 hours</u>, how many kilometers will it complete for the entire journey?

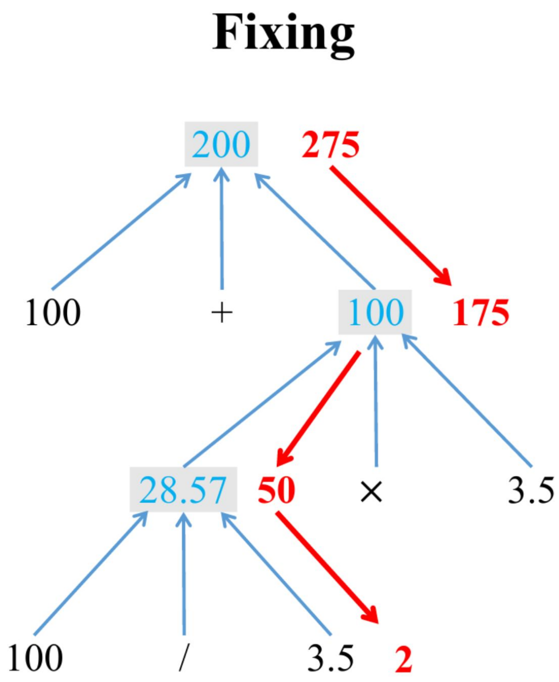


Goal-Driven Tree Structured Model

P: A truck travels 100 kilometers in 2 hours. At this speed, if it travels for another 3.5 hours, how many kilometers will it complete for the entire journey?



Learning by Fixing



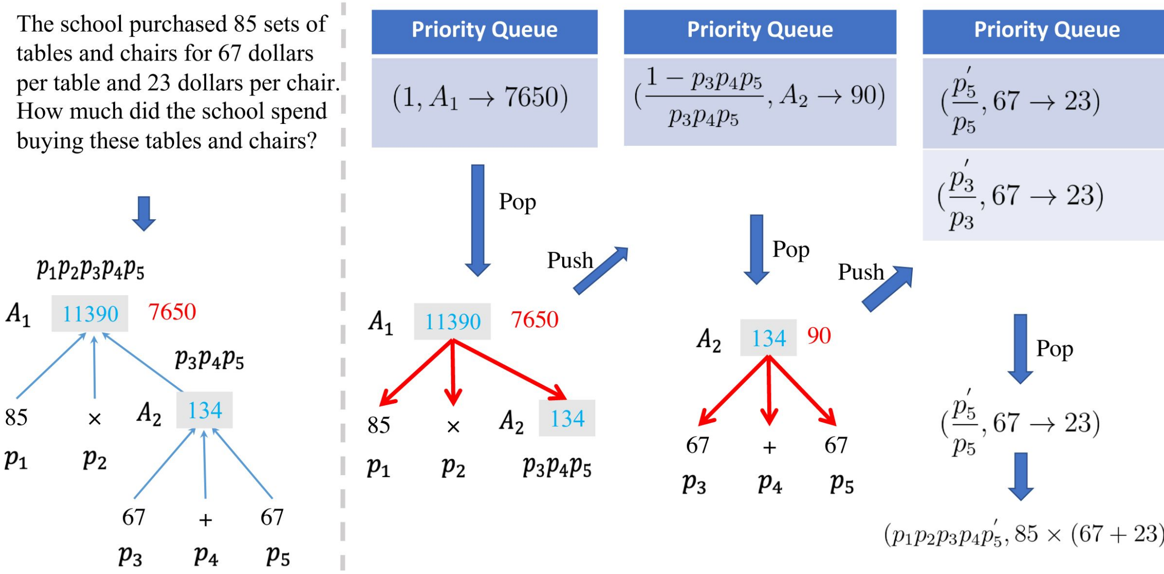
[2] Closed Loop Neural-Symbolic Learning via Integrating Neural Perception, Grammar Parsing, and Symbolic Reasoning. Qing Li, Siyuan Huang, Yining Hong, Yixin Chen, Ying Nian Wu, and Song-Chun Zhu.

Learning by Fixing

Algorithm 1 Fixing Mechanism

Input: reasoning tree $\hat{T}$, ground-truth answer y
$T^{(0)} = \hat{T}$
for $i \leftarrow 0$ to m **do**
  $T^* = 1\text{-}\textsc{Fix}(T^{(i)}, y)$
  if $T^* \neq \varnothing$ **then**
    return T^*
  else
    $T^{(i+1)} = \textsc{RandomWalk}(T^{(i)})$
return $\varnothing$

function $1\text{-}\textsc{Fix}(T, y)$
q = PriorityQueue(), S = the root node of T
$q.push(S, y, 1)$
while $(A, \alpha_A, p) = q.pop()$ **do**
  if $A \in \Sigma$ **then**
    $T^* = \hat{T}(A \rightarrow \alpha_A)$
    return T^*
  for $B \in child(A)$ **do**
    $\alpha_B = solve(B, A, \alpha_A)$
    if not ($B \in \Sigma$ and $\alpha_B \notin \Sigma$) **then**
      $q.push(B, \alpha_B, p(B \rightarrow \alpha_B))$
return $\varnothing$



Tree Regularization

$$\mathrm{Size}(T) \in [\mathrm{minSize}(T), \mathrm{maxSize}(T)]$$
$$\mathrm{minSize}(T) = a_{min}\mathrm{len}(V^{num}) + b_{min}$$
$$\mathrm{maxSize}(T) = a_{max}\mathrm{len}(V^{num}) + b_{max}$$

1. The number of operators cannot be greater than [Size(T)/2].
2. Except the last position, the number of numeric values(quantities and constants) cannot be greater than the number of operators.

$V^{num} = \{100, 2, 3.5\}$
$V^{op} = \{+, -, \times, \div, \wedge\}$
$V^{con} = \{1, 2, \pi\}$

Target size *l* = 5

$\times$	2	V^{op}
$\div$	N/A	$V^{op} \cup V^{num} \cup V^{con}$
100	1	$V^{num} \cup V^{con}$
2	1	$V^{num} \cup V^{con}$
3.5	1	$V^{num} \cup V^{con}$

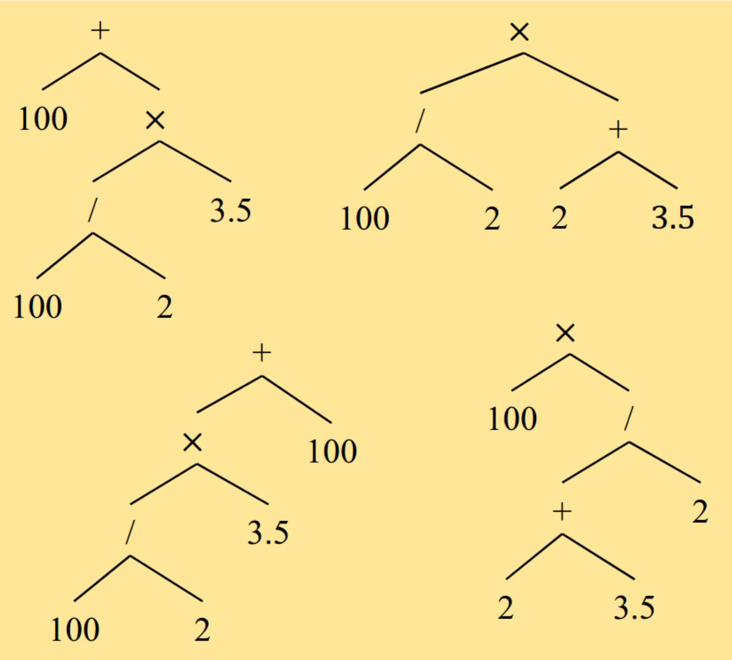
Prefix: $\times$ $\div$ 100 2 3.5

Target size *l* = 7

$\times$	2	V^{op}
$\div$	N/A	$V^{op} \cup V^{num} \cup V^{con}$
100	N/A	$V^{op} \cup V^{num} \cup V^{con}$
2	N/A	$V^{op} \cup V^{num} \cup V^{con}$
+	2	V^{op}
2	1	$V^{num} \cup V^{con}$
3.5	1	$V^{num} \cup V^{con}$

Prefix: $\times$ $\div$ 100 2 + 2 3.5

Memory Buffer



$$J(P,\beta) = -\sum_{T^* \in \beta} \log p(T^*|P)$$

Algorithm 2 Learning-by-Fixing

```
Input: training set 𝒟 = {(P_i, y_i)}_{i=1}^N
memory buffer ℬ = {β_i}_{i=1}^N, the GTS model θ
for P_i, y_i, β_i ∈ (𝒟, ℬ) do
                                                        ▷Exploring
    T̂_i = GTS (P; θ)
    T_i^* = m-FIX(T̂_i, y_i)
    if T_i^* ≠ ∅ and T_i^* ∉ β_i then
        β_i ← β_i ∪ {T_i^*}
                                                        ▷Learning
    θ = θ − ∇_θ J(P_i, β_i)
```

Experiment

- Dataset:

  Math23K, 23161 math word problems

- Evaluation Metric:

  Answer accuracies of all the top-1/3/5 predictions using beam search

- Inference Models:

  Seq2Seq, Goal-Driven Tree-Structured Model (GTS)

- Learning Strategies:

  REINFORCE, MAPO[3], LBF (Learning by Fixing), LBF-w/o-M (Fixing without Memory)

[3] Memory Augmented Policy Optimization for Program Synthesis and Semantic Parsing. Chen Liang, Mohammad Norouzi, Jonathan Berant, Quoc Le and Ni Lao.

Top-1 Answer Accuracy

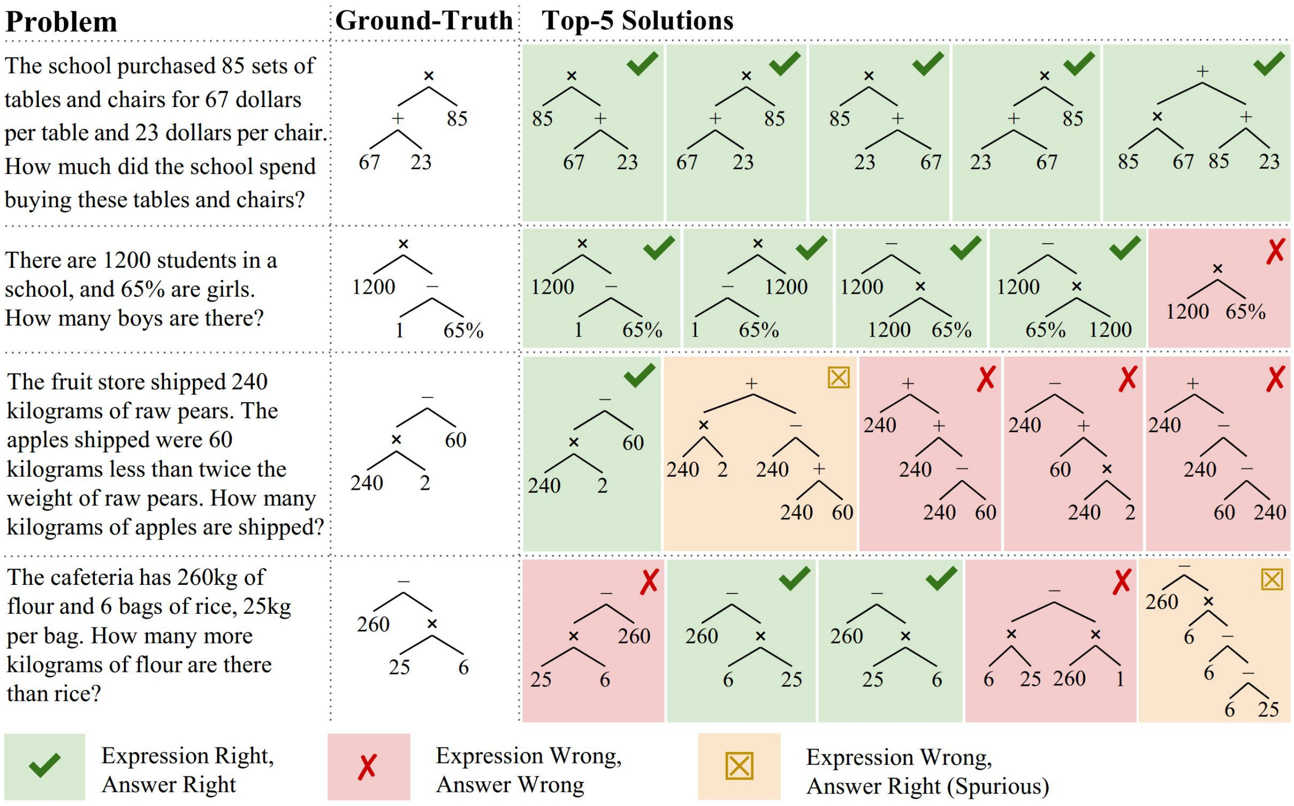
Model		Accuracy(%)
Fully-Supervised		
Retrieval (Robaidek, Koncel-Kedziorski, and Hajishirzi 2018)		47.2
Classification (Robaidek, Koncel-Kedziorski, and Hajishirzi 2018)		57.9
LSTM (Robaidek, Koncel-Kedziorski, and Hajishirzi 2018)		51.9
CNN (Robaidek, Koncel-Kedziorski, and Hajishirzi 2018)		42.3
DNS (Wang, Liu, and Shi 2017)		58.1
Seq2seqET (Wang et al. 2018)		66.7
Stack-Decoder (Chiang and Chen 2019)		65.8
T-RNN (Wang et al. 2019)		66.9
GTS (Xie and Sun 2019)		74.3
Graph2Tree (Zhang et al. 2020a)		**74.8** [1]
GTS-LBF-fully		74.1
Weakly-Supervised		
Seq2seq	REINFORCE	1.2
	MAPO	10.7
	LBF-w/o-M	44.7
	LBF	43.6
GTS	REINFORCE	15.8
	MAPO	20.8
	LBF-w/o-M	58.3
	LBF	**59.4**

Diverse Solutions with Memory Buffer, Ablative Studies

Model	Tree Size	Acc@1	Acc@3	Acc@5
Fully Supervised				
GTS		**74.3**	42.2	30.0
GTS-LBF-fully		74.1	**63.4**	**56.3**
Weakly Supervised				
GTS-LBF-w/o-M	[1,+∞)	∼0	∼0	∼0
	[2n-1,2n+1]	55.3	26.2	19.3
	[2n-1,2n+3]	58.3	27.7	20.3
	[2n-3,2n+5]	56.7	27.7	20.6
GTS-LBF	[1,+∞)	∼0	∼0	∼0
	[2n-1,2n+1]	56.7	45.3	39.1
	[2n-1,2n+3]	**59.4**	**49.6**	**45.2**
	[2n-3,2n+5]	57.6	49.3	45.2

Models \ Steps	1	10	50 (default)	100
Seq2seq-LBF-w/o-M	41.9	43.4	44.7	**47.8**
Seq2seq-LBF	43.9	**45.7**	43.6	44.6
GTS-LBF-w/o-M	51.2	54.6	**58.3**	57.8
GTS-LBF	52.5	55.8	59.4	**59.6**

Qualitative Study



Conclusions & Future Works

- We propose a weakly-supervised paradigm for learning MWPs and a novel learning-by-fixing framework to boost the learning.
- For future work, we will prevent generating equivalent or spurious solutions during training, possibly by making the generated solution trees more interpretable with semantic constraints. (See also our newest work[4]!)
- A weakly-supervised large-scale dataset on math word problems would be beneficial for this line of research.

[4] "SMART: A Situation Model for Algebra Story Problems via Attributed Grammar". Yining Hong, Qing Li, Ran Gong, Daniel Ciao, Siyuan Huang, Song-Chun Zhu.

You are welcomed to visit our project pages!

The project page of this paper: https://evelinehong.github.io/lbf-site/

For more details about the fixing mechanism: https://liqing-ustc.github.io/NGS/

For interpretable math word problems solving: https://evelinehong.github.io/smart-site/