Learning by Applying: A General Framework for Mathematical Reasoning via Enhancing Explicit Knowledge Learning

Jiayu Liu[1,2], Zhenya Huang[1,2*], Chengxiang Zhai[3], Qi Liu[1,2]

[1]Anhui Province Key Laboratory of Big Data Analysis and Application, School of Data Science & School of Computer Science and Technology, University of Science and Technology of China (USTC);
[2] State Key Laboratory of Cognitive Intelligence;
[3] University of Illinois at Urbana-Champaign

Reporter: Jiayu Liu
AAAI-2023

中国科学技术大学
University of Science and Technology of China

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

Outline

Background

Mathematical reasoning is one of the core abilities of general AI

- Requirements:
  - grasp mathematical knowledge and logical thinking from solving several mathematical problems

- Math word problems (MWP) is a fundamental reasoning task that has attracted much attention since the 1960s

Math word problem	Jack has 3 apples and Amy has 2 bananas, how many fruits do they have?
Expression: 3+2	**Answer:** 5

Background

Problem definition of MWP

- Input: a sequence of n words and numeric values $P = \{p_1, p_2, \ldots, p_n\}$
  - E.g., "Jack has 3 apples … "
- Output: mathematical expression E_P, answer S_P
  - $E_P = \{y_1, y_2, \ldots, y_m\}$, where y_i comes from $V_P = V_O \cup V_C \cup N_P$
    - ✓ V_O: operators, e.g. $\{+, \times, -, \div\}$
    - ✓ V_C: numeric constants, e.g. $\{1, \pi\}$
    - ✓ N_P: numeric variables from P, e.g. $\{3,2\}$
    - ✓ E.g., "3+2"
  - S_P: real value
    - ✓ E.g., 5

Math word problem	Jack has 3 apples and Amy has 2 bananas, how many fruits do they have?		
Expression:	3+2	**Answer:**	5

Background

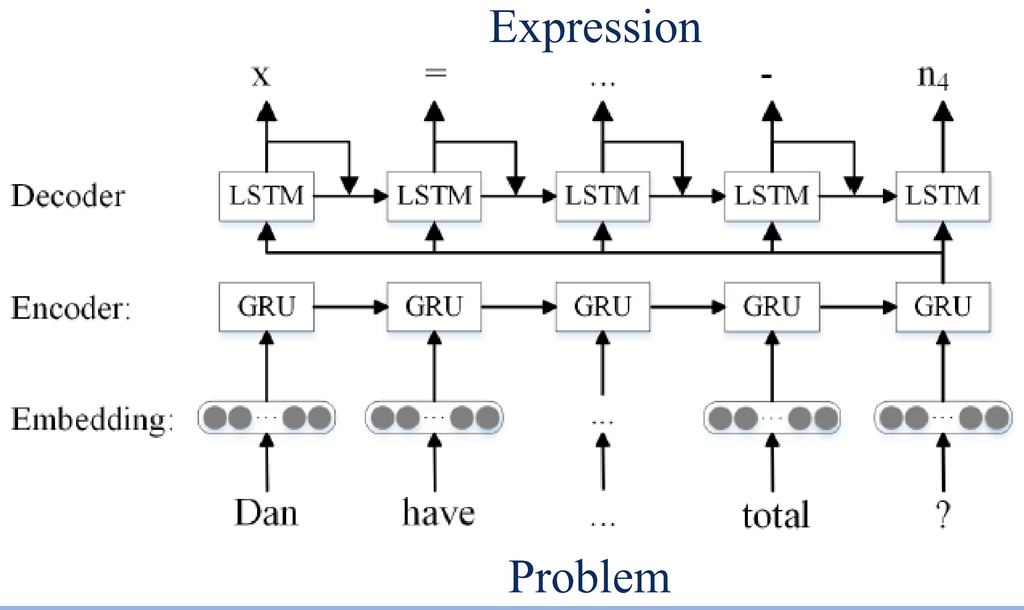
Related Work

- Traditional work
  - Rule-based Methods
  - Statistic-based Methods
  - Semantics parsing-based Methods

Tremendous human effort and low generality

- DL-based Methods
  - Seq2Seq framework: from language translation
  - Semantics-focused: Graph2Tree, HMS …
  - Reasoning-focused: GTS, TSN-MD …
  - Ensemble-based: MultiE/D

Paradigm: problem-expression

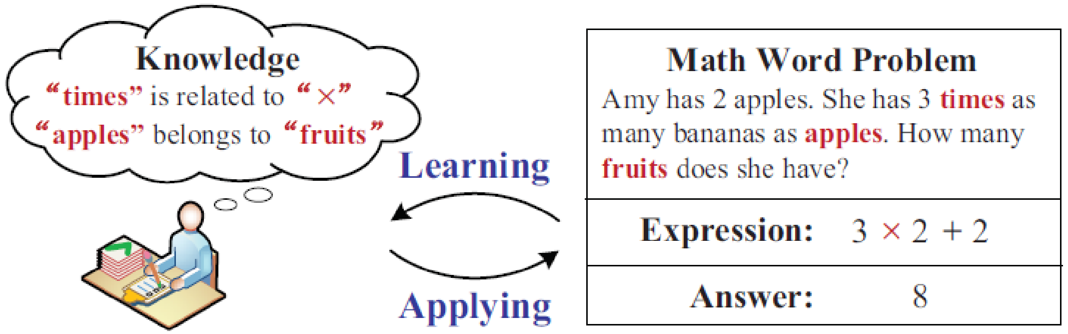


Background

Motivation

problem-expression paradigm lacks the processes of learning/applying explicit knowledge

1. Humans acquire explicit mathematical knowledge from solving problems
   - Educational theory of cognitivism (Mowrer 1960)
2. Humans produce solutions for MWP by applying this knowledge in logical thinking
3. Necessary to **integrate** these two processes into machines to build a stronger AI



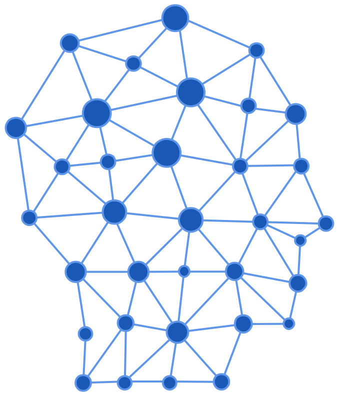
Our goal: general framework in which different MWP solvers can learn and apply explicit knowledge

Background

Challenges

- How to formalize the explicit knowledge
- How to probe a learning mechanism that simulates how humans gain knowledge from solving MWP, which meanwhile should be general to work with different solvers
- How to design a general knowledge application mechanism based on distinct solver architectures

Outline



1 Background

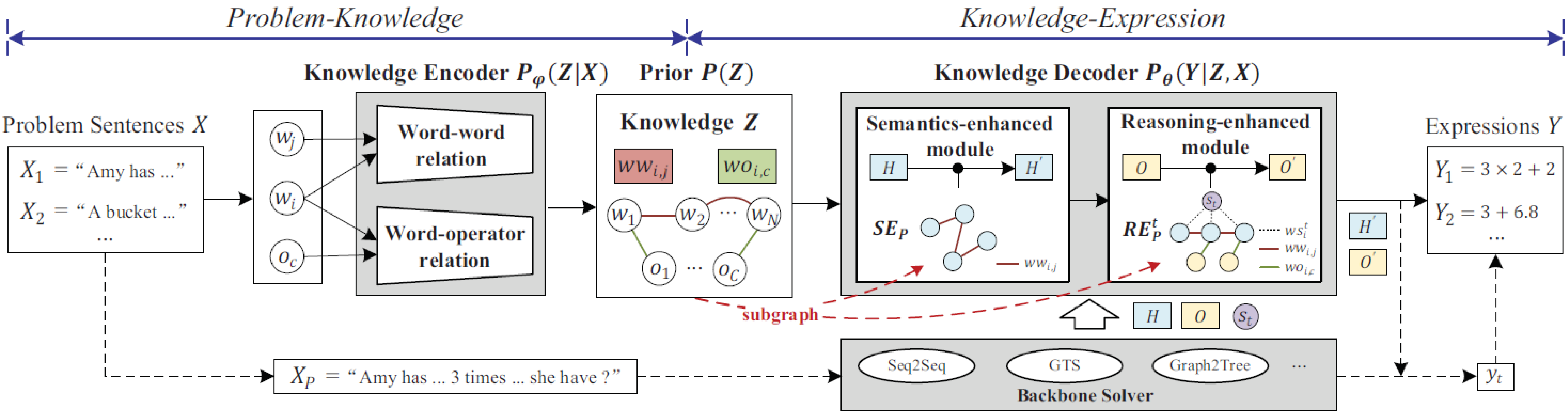
2 Our Method

3 Experiment

4 Conclusion

Our Method

LeAp: Overview

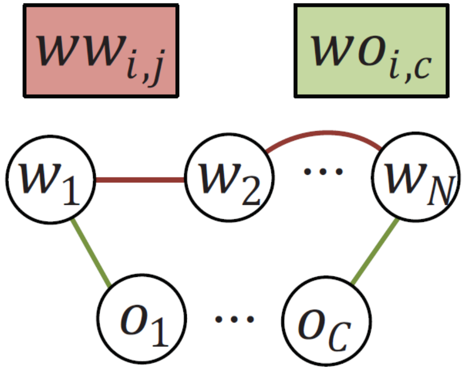


Learning by Applying (LeAp): problem-knowledge-expression paradigm

1. Learning is the process of encoding data to knowledge: problem-knowledge (Knowledge Encoder)
2. Reasoning is the process of decoding knowledge to data: knowledge-expression (Knowledge Decoder)
3. Knowledge is viewed as the hidden state (Knowledge Prior)

Our Method

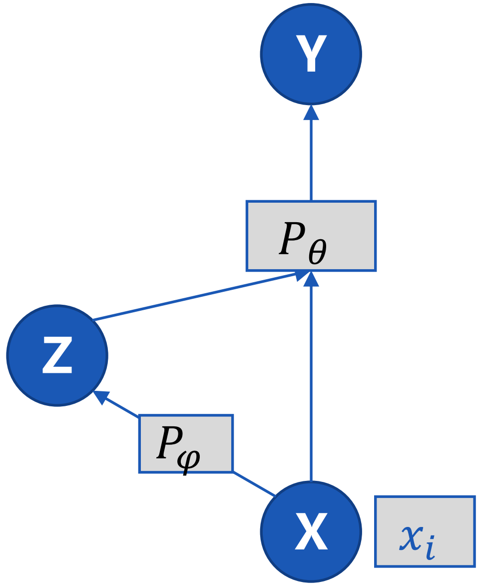
- Formulization
  - **Relation knowledge *Z*** (Bernoulli variables)
    - ✓ word-word $ww_{i,j}$
    - ✓ word-operator $wo_{i,c}$



  - **MWP**: $\max P_\theta(Y|X) = \int P(Z) \cdot P_\theta(Y|Z,X)dZ$

$$\geq \underbrace{\mathrm{E}_{P_\varphi(Z|X)}[\log P_\theta(Y|Z,X)] - \mathrm{KL}(P_\varphi(Z|X)||P(Z))}_{\text{ELBO}}$$

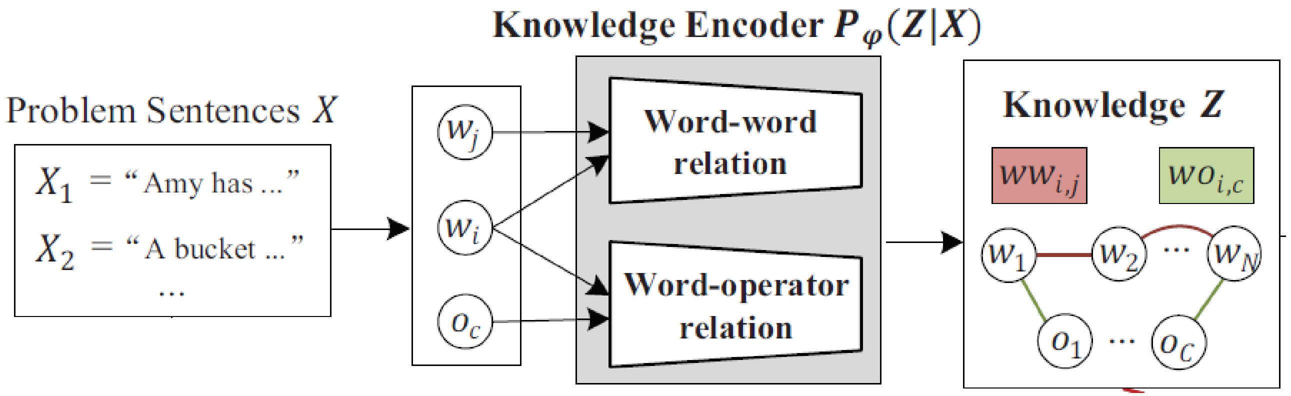
  - Three main parts:
    - ✓ $P_\varphi(Z|X)$: arbitrary NN (Knowledge Encoder)
    - ✓ $P_\theta(Y|Z,X)$: arbitrary MWP solver (Knowledge Decoder)
    - ✓ $P(Z)$: pre-defined prior (Bernoulli) distribution



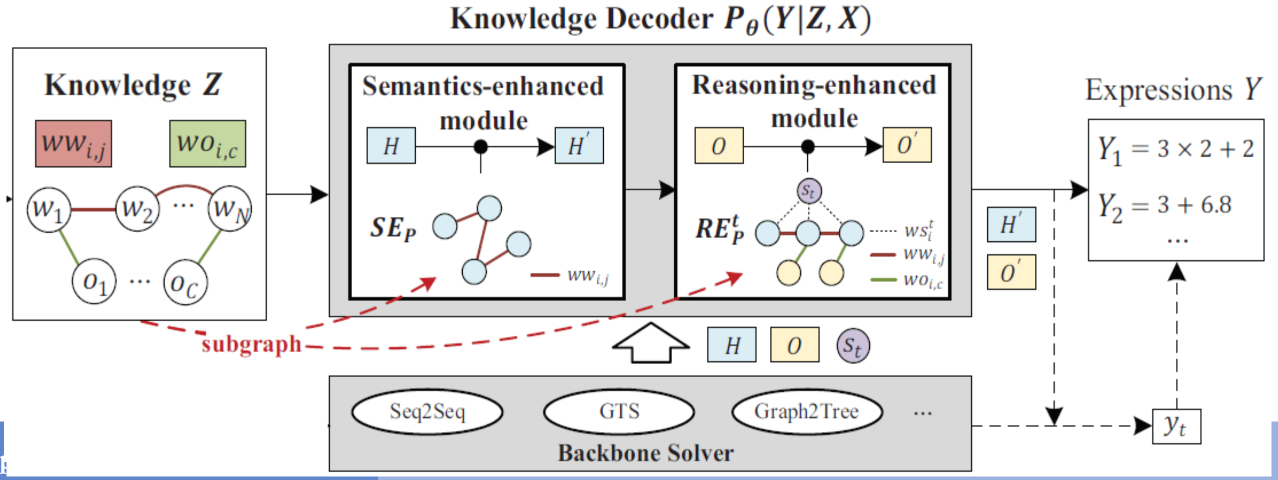
Our Method

- **Knowledge Encoder: $P_\varphi(Z|X)$**
  - $ww_{i,j} \sim Bernoulli(\sigma(f_1(w_i, w_j)))$
  - $wo_{i,c} \sim Bernoulli(\sigma(f_2(w_i, o_c)))$
- **Knowledge Decoder: $P_\theta(Y|Z,X)$**
  - Backbone Solver
  - Semantics-enhanced module
  - Reasoning-enhanced module

problem-knowledge



knowledge-expression



Our Method

- **Knowledge Decoder: $P_\theta(Y|Z,X)$**
  - Backbone Solver

$$(H, s_1) = Sol - Enc(w_i)$$

- RNN: GTS, TSN-MD
- BERT: MWP-BERT
- MWP encoder: Graph2Tree, HMS
- …

  - Backbone Solver

$$P_\theta(y_t|y_1, \dots, y_{t-1}, P) = Sol - Dec(s_t, e(y_t), options)$$
$$s_{t+1} = f_3(s_t, e(y_t), options)$$

- Output gate: Seq2Seq
- Pointer-generator network: HMS
- …

  - Discussion:
    - ✓ s_t: goal q_t in GTS, Graph2Tree,…; hidden state in Seq2Seq
    - ✓ $options$: c_t (context vector), r_t (representation of the partial expressions)

Our Method

- **Knowledge Decoder: $P_\theta(Y|Z,X)$**
  - *Semantics − enhanced Module*

$$(H, s_1) = Sol - Enc(w_i)$$

$$\mathbf{H}' = \mathbf{A_E} \cdot \mathrm{ReLU}(\mathbf{A_E} \cdot \mathbf{H} \cdot \mathbf{W}_1 + b_1) \cdot \mathbf{W}_2 + b_2$$

SE_P

Apply knowledge to improve problem comprehension

  - *Reasoning − enhanced Module*

$$P_\theta(y_t|y_1, \ldots, y_{t-1}, P) = Sol - Dec(s_t, e(y_t), options)$$

$$s_{t+1} = f_3(s_t, e(y_t), options)$$

$$\mathbf{H}^t = \mathbf{A_E} \cdot \mathrm{ReLU}(\mathbf{A_E} \cdot [ws^t \cdot \boldsymbol{s}_t, \mathbf{H}'] \cdot \mathbf{W}_4 + b_4) \cdot \mathbf{W}_5 + b_5,$$

$$\hat{\mathbf{H}}^t = \mathbf{H}^t + \mathrm{LayerNorm}(\mathbf{H}^t),$$

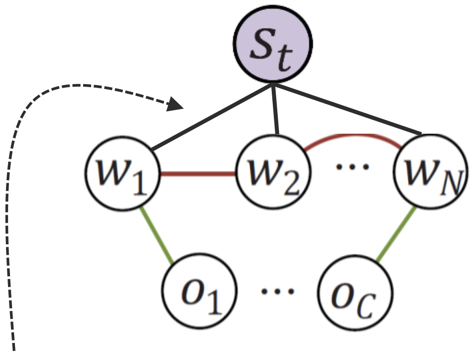
$$\mathbf{O}^t = \mathrm{LayerNorm}(\mathrm{ReLU}(\mathbf{A_D} \cdot \hat{\mathbf{H}}^t \cdot \mathbf{W}_6 + b_6)),$$

$$\hat{\mathbf{O}}^t = \mathrm{ReLU}([\mathbf{O}^t, \mathbf{O}] \cdot \mathbf{W}_7 + b_7),$$

$$e(y_t) \ni \mathbf{O}' = \mathbf{O} + \mathrm{LayerNorm}(\hat{\mathbf{O}}^t),$$

Apply knowledge to improve symbol reasoning

RE_P^t



$$ws_i^t = \frac{\exp\left(f_4\left(\boldsymbol{s}_t, \boldsymbol{h}_i'\right)\right)}{\sum_j \exp\left(f_4\left(\boldsymbol{s}_t, \boldsymbol{h}_j'\right)\right)},$$

$$f_4\left(\boldsymbol{s}_t, \boldsymbol{h}_i'\right) = \boldsymbol{v}^\top \tanh\left(\mathbf{W}_3 \cdot \left[\boldsymbol{s}_t, \boldsymbol{h}_i'\right]\right)$$

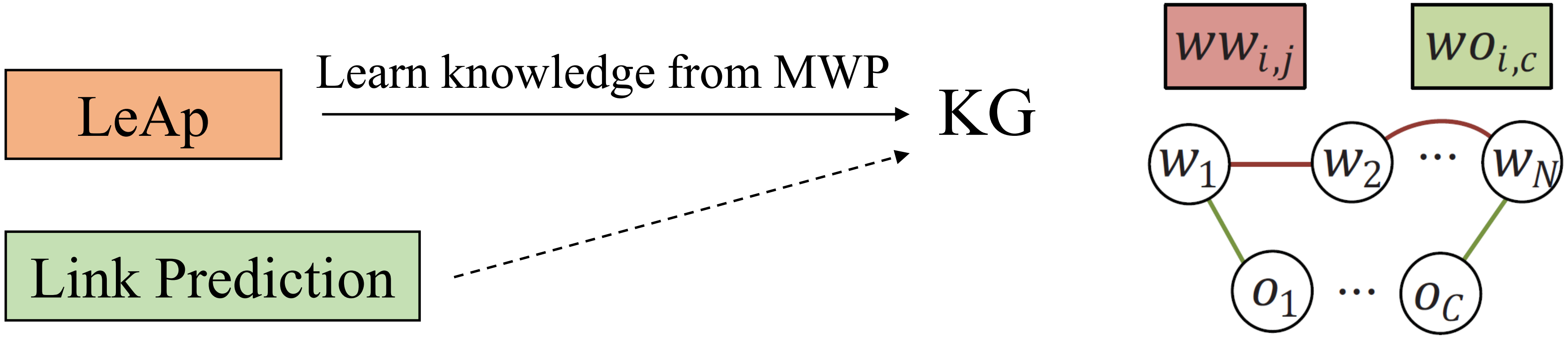
Our Method

- **Prior: $\boldsymbol{P(Z)}$**
  - $ww_{i,j} \sim Bernoulli(\delta_1)$
  - $wo_{i,c} \sim Bernoulli(\delta_1)$
  - $\delta_1 = 0.1$ for sparsity

- **Adv:** simulate the abilities of learners with knowledge backgrounds by setting $P(Z)$
  - $\delta_1 = 0.5$ for knowledge that a learner has mastered

    → Guide to learn target knowledge
  - Visualized in experiments

Theoretical Result



LeAp **VS** Link Prediction from MWP

$P(z_{i,j} = r_{i,j} | X, Y)$? $\qquad$ $P(z_{i,j} = r_{i,j} | X)$

$$P(z_{i,j} = r_{i,j} | X, Y) > P(z_{i,j} = r_{i,j} | X) \text{ ?}$$

Investigation

1) Whether the optimization objective of LeAp optimizes the posterior $P_\varphi(Z|X,Y)$

Investigation

2) If 1) holds, whether $P_\varphi(Z|X,Y)$ is larger (i.e., more accurate) than $P_\varphi(Z|X)$

Theoretical Result

- **Assumption: Effective knowledge**

Definition 1. *Effective knowledge* Z: $\forall z_{i,j} \in Z$,

$$[P_\theta(Y|z_{i,j}=1,X) - P_\theta(Y|z_{i,j}=0,X)] \cdot (2r_{i,j}-1) > 0.$$

  - If $r_{i,j} = 1$, then $P_\theta\left(Y\middle|z_{i,j} = 1, X\right) > P_\theta(Y|z_{i,j} = 0, X)$
  - If $r_{i,j} = 0$, then $P_\theta\left(Y\middle|z_{i,j} = 0, X\right) > P_\theta(Y|z_{i,j} = 1, X)$

- **Simplified setting**:

① Train φ^{LP}, X^{LP} by Link Prediction task

② Initialize LeAp with φ^{LP}, X^{LP}

③ Train LeAp, investigate how the VAE loss adjust φ^{LP}, X^{LP}

Theoretical Result

Theorem 1. Assume $P_\varphi(z_{i,j} = r_{i,j}|X) > \delta(X)$ holds in a neighborhood U of (φ^{LP}, X^{LP}) and knowledge Z is effective. Then, for each $z_{i,j} \in Z$, maximizing the objective of MWP solving in Eq. (1), i.e., $L_1 = \mathrm{E}_{P_\varphi(z_{i,j}|X)}[\log P_\theta(Y|z_{i,j}, X)]$, is equivalent to maximizing

$$L_3 = r_{i,j} \cdot P_\theta(Y|z_{i,j} = 1, X) \cdot P_\varphi(z_{i,j} = 1|X) + (1 - r_{i,j}) \cdot P_\theta(Y|z_{i,j} = 0, X) \cdot P_\varphi(z_{i,j} = 0|X) \quad (12)$$

in U, where $\delta(X) \triangleq \max \{\frac{1}{1+\frac{\beta(1-r_{i,j}, r_{i,j}, c)}{\beta(r_{i,j}, 1-r_{i,j}, c)}}\}|_{c=\theta, \boldsymbol{x}_i \in X}$, and $\beta(a,b,c) \triangleq P_\theta(Y|z_{i,j} = a, X) \cdot \|\frac{\partial P_\theta(Y|z_{i,j}=b,X)}{\partial c}\|$.

Answer

Investigation

1) Whether the optimization objective of LeAp optimizes the posterior $P_\varphi(Z|X,Y)$

Theorem 2. Under the assumption of "Effective knowledge", the following inequality holds:

$$\frac{P(Y|z_{i,j} = r_{i,j}, X) \cdot P(X)}{P(X,Y)} > 1. \quad (17)$$

Answer

Investigation

2) If 1) holds, whether $P_\varphi(Z|X,Y)$ is larger (i.e., more accurate) than $P_\varphi(Z|X)$

Theoretical Result

- **Summary**
  - Under the Assumption of Effective Knowledge
  - Theorem1:

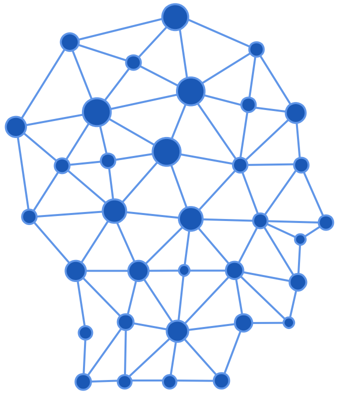
    LeAp is optimizing $P(z_{i,j} = r_{i,j} | X, Y)$ √

  - Theorem2:

    $P(z_{i,j} = r_{i,j} | X, Y) > P(z_{i,j} = r_{i,j} | X)$ √

  - Note: superiority lies in the mechanism to calculate the posterior probability based on the information (i.e., Y) provided by solving MWP

Outline



Experiment

Setups

- Dataset
  - Math23K, MAWPS, SVAMP
- Baseline methods
  - Seq2Seq
  - Graph2Tree
  - HMS
  - GTS
  - TSN-MD
  - Multi-E/D

Graph2Tree, HMS → Semantics-focused methods

GTS, TSN-MD → Reasoning-focused methods

Multi-E/D → Ensemble-based method

- Evaluate metric: Answer Accuracy

Experiment

Accuracy Performance

directly apply knowledge Z from external knowledge bases (HowNet & ConceptNet) in Knowledge Decoder

	Math23K			MAWPS			SVAMP		
	ORI	LeAp	LeAp-EK	ORI	LeAp	LeAp-EK	ORI	LeAp	LeAp-EK
Seq2Seq	0.640	**0.660****	0.652**	0.797	0.803	**0.807***	0.200	**0.236*****	0.220***
Graph2Tree	0.774	0.779*	**0.782****	0.837	**0.852****	0.849**	0.319	**0.341*****	0.325*
HMS	0.761	**0.769**	0.765	0.803	**0.812***	0.805	0.179	**0.196****	0.191**
GTS	0.756	**0.772****	0.767**	0.826	**0.834****	0.830*	0.277	**0.285**	0.279
TSN-MD	0.774	**0.786****	0.778*	0.844	**0.853***	0.848	$0.290^{\dagger}$	**0.302****	0.294*
Multi-E/D	0.784	0.791*	**0.793****	/	/	/	/	/	/

Table 1: Answer accuracy ($***: p \leq 0.001$, $**: p \leq 0.01$, $*: p \leq 0.05$). †: implemented by MTPToolkit (Lan et al. 2022).

- LeAp can enhance the mathematical reasoning ability of MWP solvers
- Applicability of Knowledge Decoder that applies knowledge for different solvers and further verifies the reasonability of our "Effective knowledge" assumption
- Reflects the importance and benefits of LeAp's learning mechanism to gain knowledge autonomously

Experiment

Accuracy Performance

LeAp (backbone)	Math23K		MAWPS		SVAMP	
	w/o SE	w/o RE	w/o SE	w/o RE	w/o SE	w/o RE
Seq2Seq	0.645	0.648	0.798	0.802	0.210	0.228
Graph2Tree	0.778	0.776	0.846	0.845	0.335	0.328
HMS	0.766	0.762	0.809	0.801	0.193	0.189
GTS	0.770	0.761	0.830	0.830	0.265	0.284
TSN-MD	0.782	0.783	0.847	0.852	0.293	0.300
Multi-E/D	0.790	0.788	/	/	/	/

- Effectiveness of each component
- Semantics/Reasoning-enhanced modules are suitable for different types of backbones

Experiment

Knowledge Learning

LeAp (backbone)	word-word pairs	word-operator pairs
Seq2Seq	house-home egg-food add-all	total-“+” selling-“−” pieces-“×”
Graph2Tree	apple-fruit sale-buy give-hold	more-“+” times-“×” gave-“−”
HMS	person-people more-total more-another	leftover-“+” other-“+” add-“+”
GTS	potato-food apple-fruit piece-part	earned-“+” rate-“÷” added-“+”
TSN-MD	per-each store-sale red-blue	give-“−” costs-“−” borrowed-“+”

- LeAp gains reasonable and interpretable knowledge with different backbones

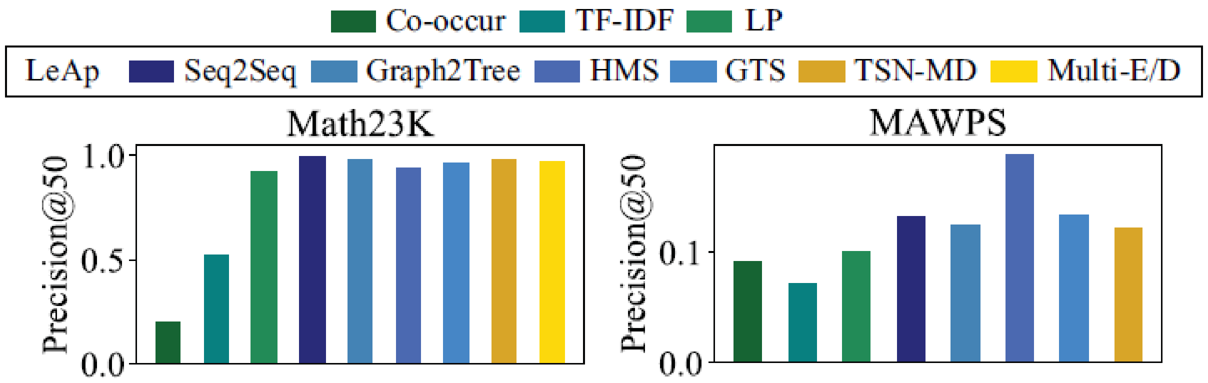


Figure 3: $Precision@50$ of word-word relations $ww_{i,j}$.

- Superiority and robustness of LeAp's autonomous learning mechanism
- Rationality of our theoretical analyses.

Experiment

Knowledge Prior

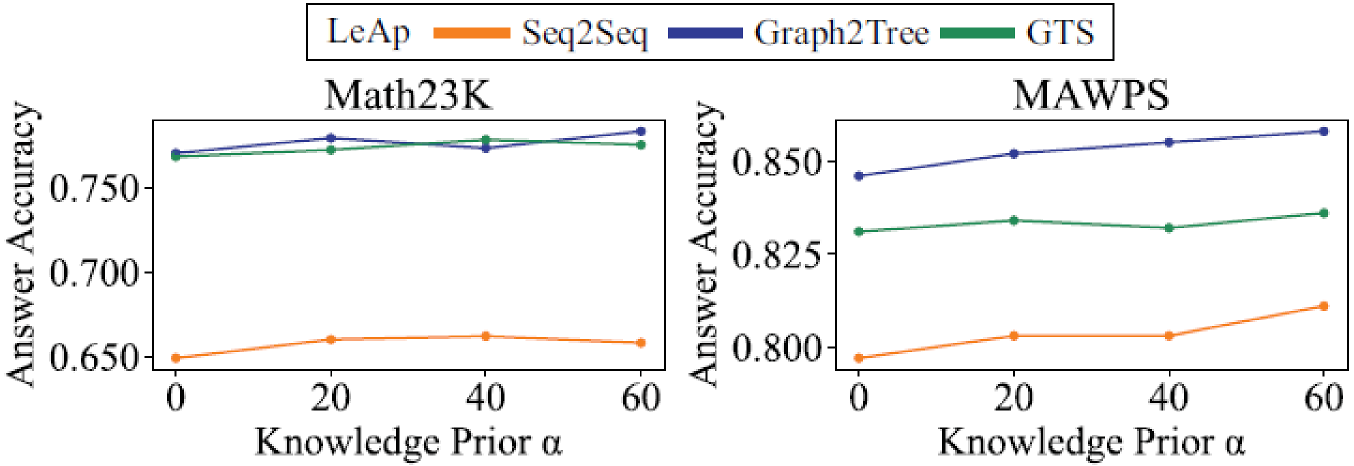
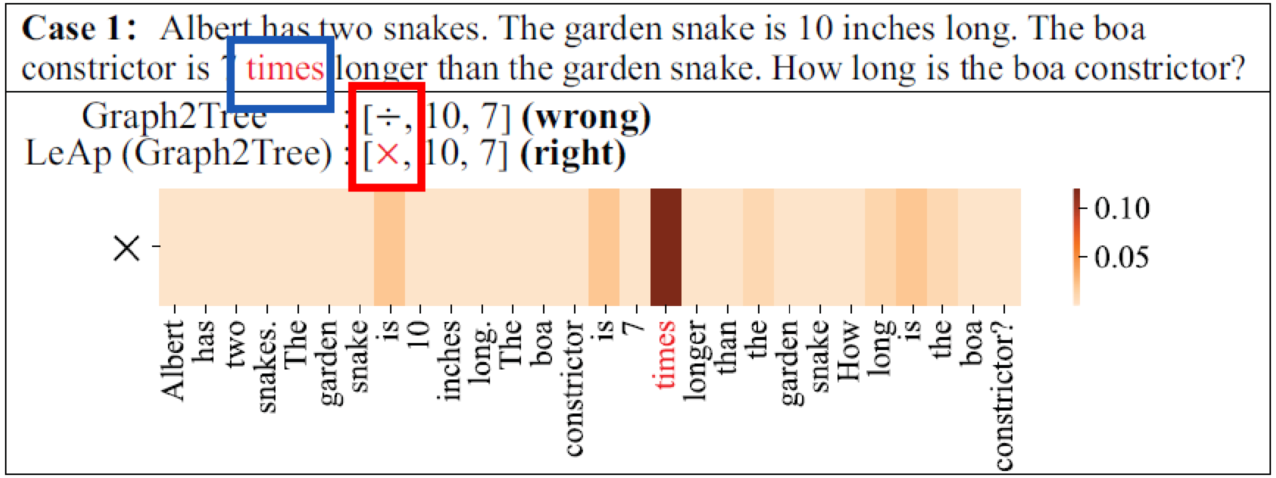


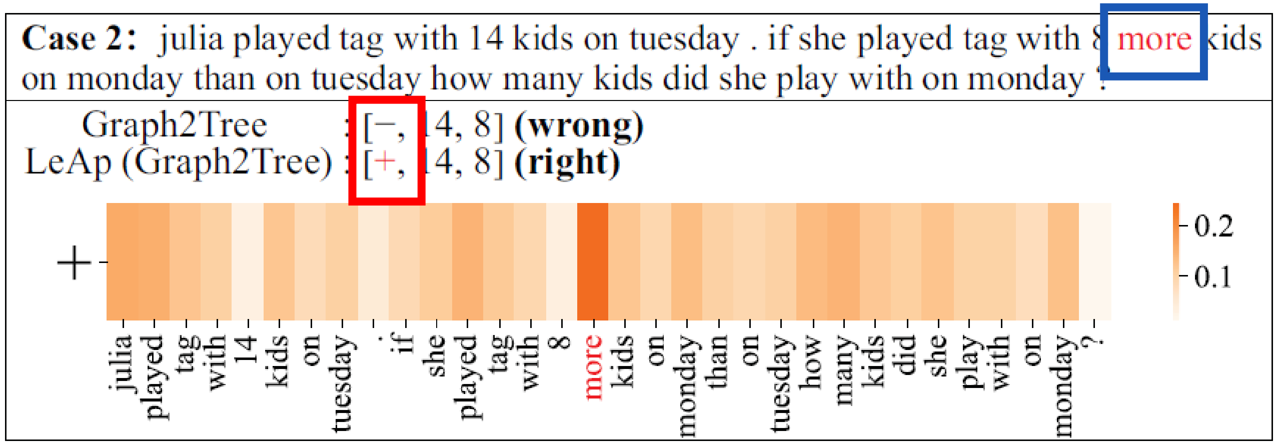
Figure 4: Answer accuracy with $\alpha = 0\%, 20\%, 40\%, 60\%$.

- With the increase of α, the answer accuracy of LeAp shows an increasing trend, just as a human learner with a richer knowledge foundation can better carry out mathematical reasoning
- When α = 0%, LeAp still outperforms the original backbones (“ORI”)

Experiment

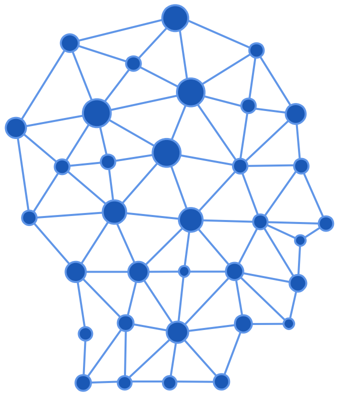
Case study





- By applying the learned knowledge $wo_{i,c}$ between **“times” and “×”, “more” and “+”**, LeAp reasons **“×”** and **“+”** accurately
- Can also explain how to reason the corresponding answers

Outline



1 Background

2 Our Method

3 Experiment

4 Conclusion

Conclusion

Summary

- Learning by Apply (LeAp) for explicit knowledge learning and applying
  - General problem-knowledge-expression paradigm
  - Semantics/reasoning-enhanced modules to strengthen problem understanding and symbol reasoning by applying knowledge effectively
  - Theoretically proved the superiority of LeAp's autonomous learning mechanism
- Experimental results proved the effectiveness and interpretability

Future Work

- Extend to other kinds of knowledge/problems
- Explore its potential to enrich external knowledge bases
- …

Thanks for your listening!

For more details, please refer to our paper!

Reporter: Jiayu Liu
jy251198@mail.ustc.edu.cn

中国科学技术大学
University of Science and Technology of China

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN