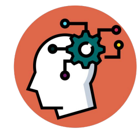


Neuro-symbolic Reasoning in Modern AI

Eric Chen
zeming.chen@epfl.ch

nlp.epfl.ch

Outline

- Background
- Neural-symbolic joint reasoning
- Symbolic reasoning controller
- Symbolic guidance for large neural models

Background

Machines that Understand Human

[Grouping Game] Passage:
Seven directors -A, B, C, D, E, F, and G- serves on the X committee or the Y committee.

If A serves on X, then B serves on Y. **R-1**
If C serves on X, then D and E serve on Y. **R-2**
F serves on a different committee with G. **R-3**
E serves on a different committee with A. **R-4**
If G serves on X, so does B. **R-5**
Rules

Question:
If D and F both serve on the X committee, **Fact**
then which one of the following could be true?

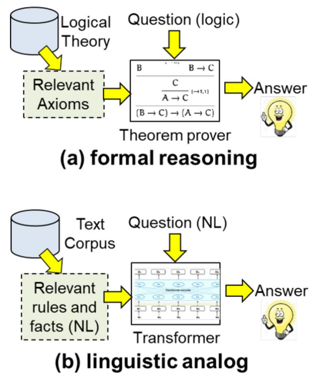
Options:
A. A and C both serve on the X committee.
$(C\ on\ X)\&(D\ on\ X)$ confict with R-2
B. A and E both serve on the Y committee.
$(A\ on\ Y)\&(E\ on\ Y)$ confict with R-4
C. B and G both serve on the X committee.
$(G\ on\ X)\&(F\ on\ X)$ confict with R-3
D. C and E both serve on the Y committee. √
E. G and E both serve on the X committee.
$(G\ on\ X)\&(F\ on\ X)$ confict with R-3

Participants	**Positions**	**Fact**
(A, B, C, D, E, F, G)	(X, Y)	$(D\ on\ X)\&(F\ on\ X)$

Rules to Logical Expressions
R-1: $A\ on\ X \rightarrow B\ on\ Y$
R-2: $C\ on\ X \rightarrow (D\ on\ Y)\&(E\ on\ Y)$
R-3: $Position\ of\ F \neq Position\ of\ G$
R-4: $Position\ of\ E \neq Position\ of\ A$
R-5: $G\ on\ X \rightarrow B\ on\ X$

AR-LSAT: Investigating Analytical Reasoning of Text (Zhong et al., 2021)

Symbolic vs. Neural

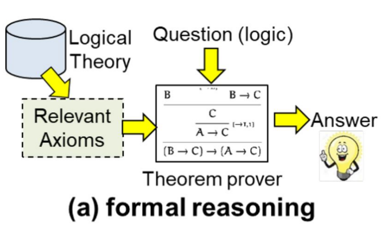


Transformers as Soft Reasoners over Language (Clark et al., 2020)

Symbolic vs. Neural

Traditional AI: symbolic systems

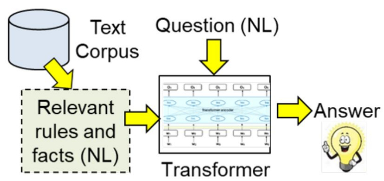
- Parser, automatic theorem proving, symbolic regression, etc.
- Pros: Explainable, trustworthy, precise
- Cons: Not expressive, difficult to scale up, not trainable



Symbolic vs. Neural

Modern AI: Deep-learning-based models

- RNN, Transformer, Graph-NN, etc.
- Pros: high learnability through differential learning, can handle inputs in various formats, domains, even muldoles, very expressive, cheap (relatively)
- Cons: hard to interpret, not transparent, exploit artifacts and bias.



(b) linguistic analog

Neural-symbolic AI

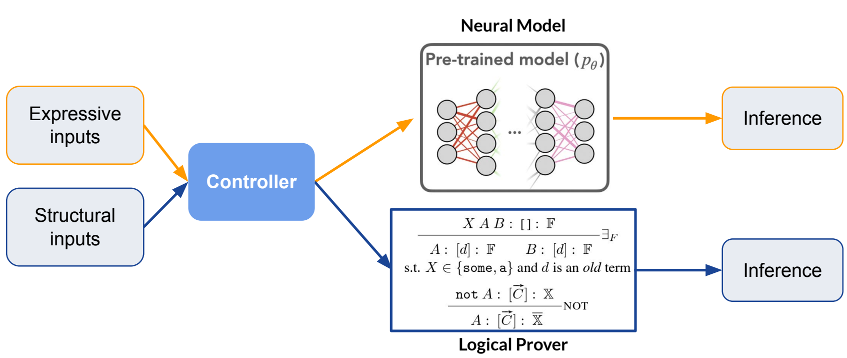
Can we combine neural and symbolic methods to achieve more complex reasoning?

[Natural Logic meets Machine Learning](#)
Workshop @IWCS 2021

Neural-symbolic Joint Reasoning

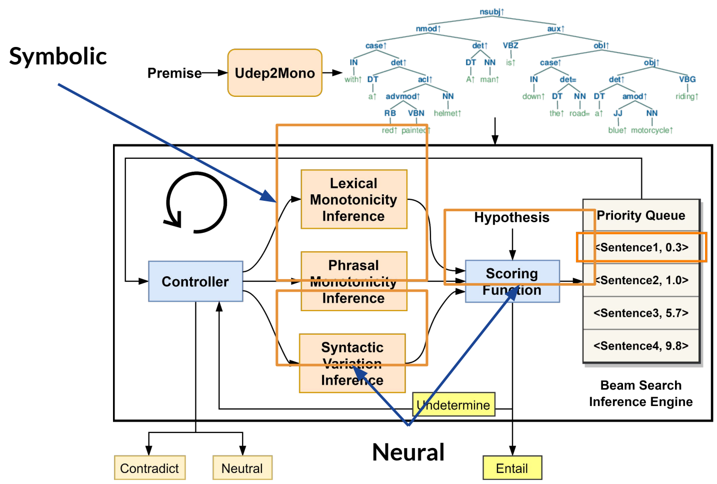
Neural-symbolic Joint Reasoning

- Joint reasoning by combining symbolic and neural components
- Apply symbolic or neural model based on the current situation

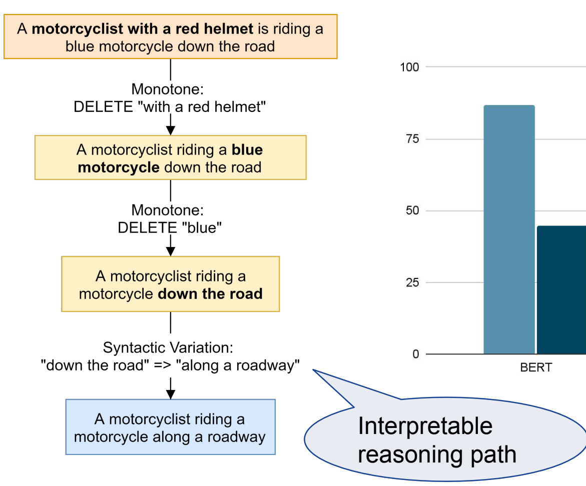


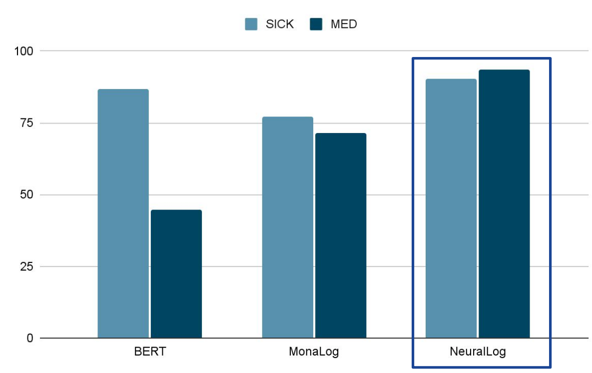
Neural-symbolic Joint Reasoning: NeuralLog

NeuralLog: Natural Language Inference with Joint Neural and Logical Reasoning (Chen et al., 2021)



NeuralLog





Summary 1

Joint reasoning allows neural and symbolic modules to solve parts of the problem they are good at

Advantages

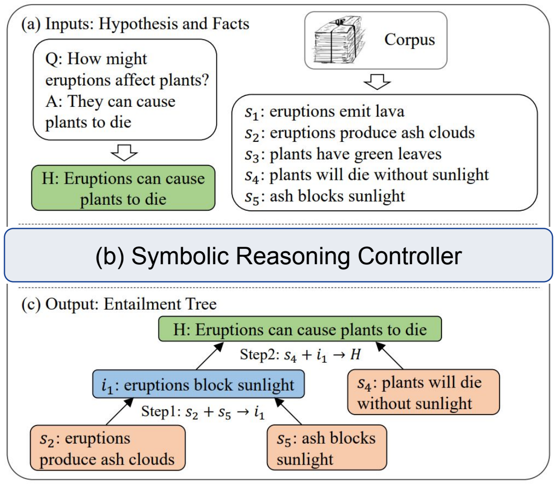
- Combines the advantages of neural and symbolic models
- Can provide a clear reasoning path for explanation
- Model-agnostic, can evolve through time

Limitations

- Require custom modules for different types of reasoning
- Computationally can be inefficient
- Large error propagation

Symbolic Reasoning Controller

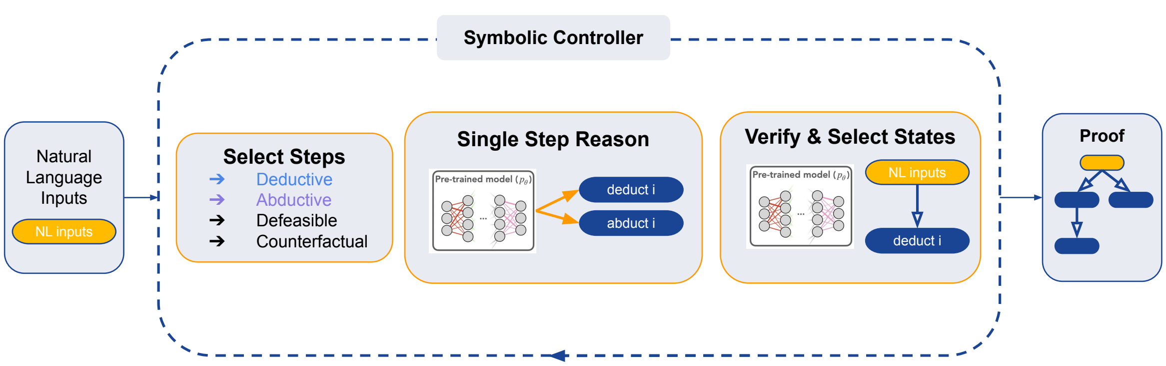
Symbolic Reasoning Controller



- Generating entailment tree to prove a hypothesis with given facts
- Highly complex, very challenging (even for humans)
- Can neural-symbolic method solve this task? Which part should be handled by symbolic methods, and which by neural models?

Symbolic Reasoning Controller

- Build symbolic reasoning on top of neural models
- Materialize internal knowledge of neural models



Symbolic Reasoning Controller: Entailer

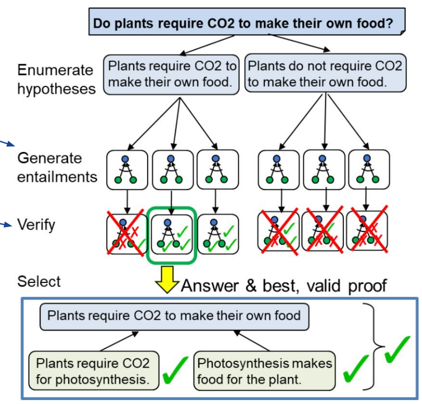
Entailer: Answering Questions with Faithful and Truthful Chains of Reasoning (Tafjord etal., 2022)

One-step backward chainer

- **Generate steps (neural)**
  - **H → P**: Given a hypothesis H, generate a set of premises P that may entail H
- **Verify steps (neural)**
  - **H → Sd**: Score the truthfulness of hypothesis H (or premise pi)
  - **P H → Se**: Score validity of a candidate entailment (P, H)

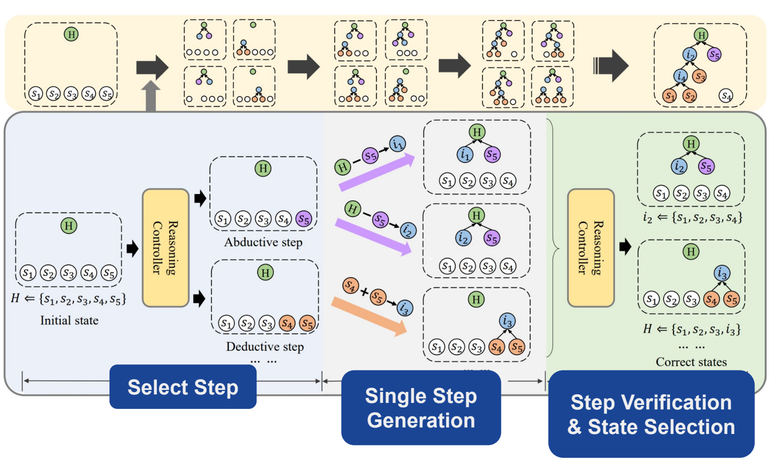
Backward Chaining (symbolic)

- Start with the H and iteratively expands
- Search for a sub-tree with Se > Sd
- Prune sub-trees with Se < Sd



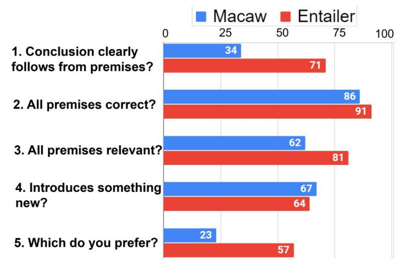
Symbolic Reasoning Controller: METGEN

METGEN: A Module-Based Entailment Tree Generation Framework for Answer Explanation (Hong et al., 2022)



Symbolic Reasoning Controller

- Entailer performs much better under Human Evaluation
- METGEN outperforms baselines on automatic and manual metrics
- Neural-symbolic systems (entailer, metegen) are better than pure neural models (Macaw, T5)



Method	Task1 Automatic	Task1 Manual	Task2 Automatic	Task2 Manual
EntialmentWriter (T5-large)	35	46	21	26
EntialmentWriter-Iter (T5-large)	35	47	25	35
METGEN-prefixed (Ours)	**36**	**53**	**27**	**39**

Summary 2

Symbolic reasoning controller wraps neural models in a human-defined symbolic algorithm

Advantages

- ❖ Human-defined, programmable reasoning process
- ❖ Highly explainable, clear proof
- ❖ Adaptable, model agnostic, can evolve in the future

Limitations

- ❖ Require human efforts for customized controller
- ❖ Error propagation from different modules
- ❖ Computationally inefficient

Symbolic Guidance for Large Neural Models

Prompt & In-context Learning

Large Language Models (LLMs) can do **in-context few-shot learning** via **prompting** ([Liu et al., 2021])

- Industry provides many state-of-the-art LLMs (66B-540B parameters):
  - OpenAI: GPT-3 (Brown et al., 2020), Instruct GPT (Ouyang et al., 2022)
  - Google: PaLM (Chowdhery et al., 2022), LaMDA (google research)
  - Meta: OPT (Meta AI)
- **Prompting**: using natural language instructions to manipulate model behavior
  - A product from **pre-training** on massive amount of text data
- **In-context learning**: append a few examples in the prompt as demonstrations
  - Model can simply learn from the demonstrations without gradient-based learning
  - Very effective results, especially on large models

Question: Can we guide LLMs to do symbolic reasoning via prompting?

Chain-of-thought Prompting

Chain-of-Thought Prompting Elicits Reasoning in Large Language Models ([Wei et al., 2022](#))

- ❖ Guide LLMs to generate **intermediate reasoning steps** through prompt.
- ❖ Require carefully-crafted and task-specific step-by-step reasoning examples

- ▯ Standard Few-Shot

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?
A: The answer is 11.

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A:

(Output) The answer is 8. ✗

- ▯ Few-Shot-CoT

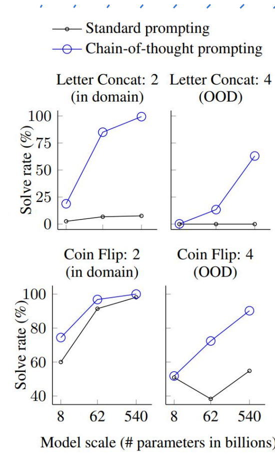
Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?
A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. 5 + 6 = 11. The answer is 11.

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A:

(Output) The juggler can juggle 16 balls. Half of the balls are golf balls. So there are 16 / 2 = 8 golf balls. Half of the golf balls are blue. So there are 8 / 2 = 4 blue golf balls. The answer is 4. ✓

Chain-of-thought Prompting

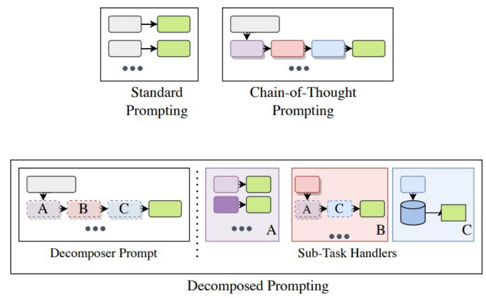
- CoT outperforms standard prompting on symbolic tasks
- CoT shows better OOD performance, more generalizable
- Guide LLMs to generate reasoning steps unlocks the symbolic ability of LLMs



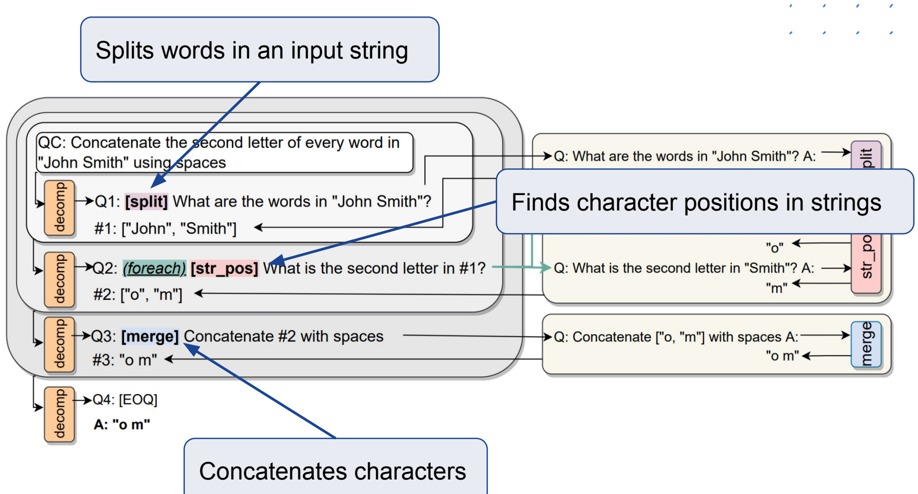
Decomposed Prompting

Decomposed Prompting: A Modular Approach for Solving Complex Tasks ([Khot et al., 2022](https://arxiv.org/abs/2210.02406))

- ❖ **Decomposed reasoning**: decompose complex tasks into simpler sub-tasks.
- ❖ Guide LLMs to decompose complex reasoning into multiple easier subtasks.
- ❖ Iteratively calls the decomposer prompt to generate the next question and sub-task at each step
- ❖ Each sub-task handled by sub-task specific handlers using various methods (neural or symbolic)

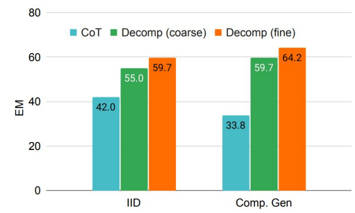


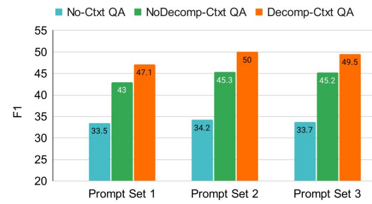
Decomposed Prompting



Decomposed Prompting

- DecomP shows significant improvement over baselines
- Fit for complex reasonings: long-context, open-domain question answering
- Better performance than CoT prompting
- An effective way of guiding LLMs to perform symbolic reasoning via simple prompting





Summary 3

We can guide Large language models to perform symbolic reasoning via prompting

Advantages

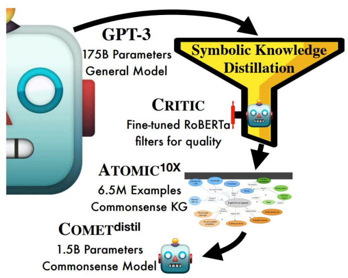
- Few-shot even zero-shot symbolic reasoning
- No additional training needed
- Reasoning process is transparent and explainable
- Composable for large reasoning systems

Limitations

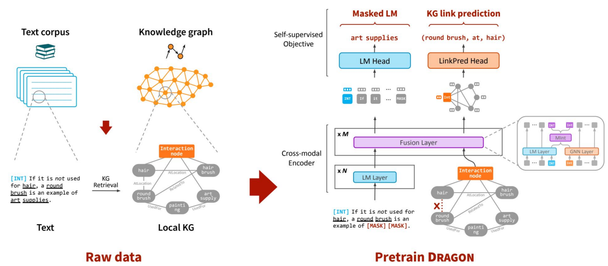
- Reasoning process is not consistent, can be noisy
- Reasonings are not fully trustworthy
- Require access to LLMs, cost
- Low controllability, over-reasoning

What's Next: Knowledge

- Large Language Models as knowledge engines
- Knowledge graph integration in NLP systems



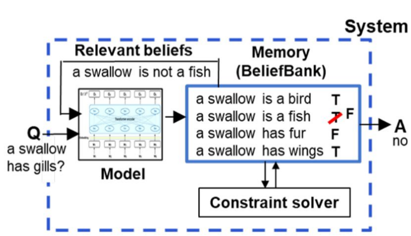
Symbolic Knowledge Distillation: from General Language Models to Commonsense Models ([West et al., 2021](#))



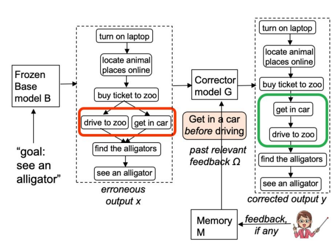
Deep Bidirectional Language-Knowledge Graph Pretraining ([Yasunaga et al., 2022](#))

What's Next: Beliefs

- Are models' decisions consistent with their internal beliefs?
- Explicitly track model's beliefs for verification and correction



BeliefBank: Adding Memory to a Pre-Trained Language Model for a Systematic Notion of Belief ([Kassner et al., 2021](#))



Learning to Repair: Repairing model output errors after deployment using a dynamic memory of feedback ([Tandon et al., 2022](#))

What's Next: Learning

- Design loss functions to calibrate models based on logical constraints
- Train models to be accurate and logically consistent to human reasoning
- Ex: Learning counterfactual reasoning from counterfactual constraints

Counterfactual constraints

Given **(p,h,l)** and new **(p',l')**

$$\forall p, p', h.\ \underbrace{l(p,h) \wedge \mathrm{Counter}(p,p',h) \rightarrow l'(p',h) \wedge l \neq l'}_{\textbf{reversal}\text{:counterfactual instance should switch label}}$$

$$\forall p, p', h.\ \underbrace{l(p,h) \wedge \mathrm{Invariant}(p,p',h) \rightarrow l'(p',h) \wedge l = l'}_{\textbf{preserve}\text{:invariant examples should preverse label}}$$

My current work (in progress)

Concluding Thoughts

- Neural models and symbolic systems both have their advantages and limitations

- Combining them as **neural-symbolic** systems has shown effective results on solving **complex reasoning** and provide **explainable** thought process

- Current methods still operates on a system level
  - Can we embed symbolic scaffolds into neural models to help them learn human-like behaviors directly?

Additional Paper List

- ❖ Symbolic guidance

Maieutic Prompting: Logically Consistent Reasoning with Recursive Explanations

Large Language Models are Zero-Shot Reasoners

- ❖ Logical Constraint Learning

A Semantic Loss Function for Deep Learning with Symbolic Knowledge

Deep Learning with Logical Constraints

- ❖ Knowledge Integration with Neural Models

QA-GNN: Reasoning with Language Models and Knowledge Graphs for Question Answering

GreaseLM: Graph REASoning Enhanced Language Models for Question Answering

Thank You