AIMS STEM Education

https://www.aimspress.com/journal/steme

STEM Education, 3(4): 251–262
DOI:10.3934/steme.2023015
Received: 21 August 2023
Revised: 19 October 2023
Accepted: 02 November 2023
Published: 15 November 2023

Research article

Reasoning arithmetic word problems entailing implicit relations based on the chain-of-thought model

Hao Meng[1], **Lin Yue**[2], **Geng Sun**[3,4] **and Jun Shen**[5,*]

[1] School of Computing and Information Technology, University of Wollongong, Wollongong, Australia; hm578@uowmail.edu.au

[2] Department of Actuarial Studies and Business Analytics, Macquarie University, Sydney, Australia; lin.yue@mq.edu.au

[3] School of Computer Engineering, Chongqing College of Humanities, Science and Technology, Chongqing, China

[4] Vermilion Cloud, Sydney, Australia; gsun@vermilioncloud.com.au

[5] School of Computing and Information Technology, University of Wollongong, Wollongong, Australia; jshen@uow.edu.au

* **Correspondence:** Email: jshen@uow.edu.au; Tel: +61 2 42213873.

Academic Editor: Ergun Gide

Abstract: Solving arithmetic word problems ($AWPs$) that involve deep implicit relations can be quite challenging. However, the paper proposed two approaches to tackle this issue. The first approach used the modifier-to-matrix (MTM) model to extract noun modification components from the problem text. Specifically, a missing entity recovery (MER) model translated explicit expressions into a node dependency graph (NDG). The nodes on the graph then recursively acquired connections from the knowledge base through the MER model until the goal was achieved with a known quantity. The solving engine then selected the appropriate knowledge as the prompt. The second approach proposed in the paper was a comprehensive one that combined explicit and implicit knowledge to enhance reasoning abilities. The experimental results of the dataset demonstrate that the proposed algorithm is superior to the baseline algorithms in solving $AWPs$ that require deep implicit relations.

Keywords: arithmetic word problem (AWP), implicit relation, chain-of-thought, missing entity recovery (MER), prompt

1. Introduction

For decades, the challenge of automatically solving arithmetic word problems (AWPs) has been a struggle for artificial intelligence. However, in recent years, deep learning methods have emerged as powerful tools for improving success rates in this area [1]. Deep learning has improved natural language processing tasks, such as question-answering and machine translation [2, 3], and has also made significant progress in mathematical reasoning [4–7]. However, current deep-learning solvers focus primarily on solving problems rather than teaching step-by-step reasoning. Language models have shown great promise in mathematical reasoning, but there are concerns regarding the accuracy of their outputs. Researchers have noted the potential for generating ungrounded answers [8], and users often have to verify predicted outcomes with extra effort. While recent prompting strategies have been developed to provide rationales before making predictions [7], language models can still produce hallucinated statements and wrong answers. Therefore, there is a pressing need for novel approaches that enable more reliable reasoning.

Earlier research on the development of reasoning chains, conducted by [7], or relied on a solitary human-annotated prompt. However, manually constructing reasoning chains has two primary drawbacks. First, current models may need the capability to learn and execute all necessary reasoning steps as tasks become more complex. Additionally, these models may need help in generalizing different tasks. Second, depending on a single decoding process leaves it susceptible to incorrect inference steps, which can result in an incorrect final answer. To overcome these limitations, recent studies have focused on two primary approaches. The first approach is process-based, which involves creating more intricate demonstrations by hand. Zhou et al. [9] and Chen et al. [10] are examples of studies that have used this approach. The second approach is outcome-based, which involves utilizing ensemble-like methods. Wang et al. [11] and Li et al. [12] are examples of studies that have adopted this approach.

The study presents a qualia-based template prompt for the modifier-to-matrix (MTM) model, inspired by a qualia structure [13] to represent the derived properties. The study is unique in that it utilizes implicit mathematical knowledge through its reasoning approach, a general framework based on a state-action paradigm. Additionally, the paper introduces the missing entity recovery (MER) model, resulting in improved solution accuracy and reasoning interoperability. This groundbreaking approach combines explicit and implicit knowledge learning to enhance reasoning abilities, making it possible to comprehend complex problems and develop more effective solutions.

2. Related work

2.1. Mathematical trustworthy reasoning

Large language models (LLMs) are incredibly powerful in modeling natural language, but they can face significant obstacles when considering mathematical reasoning. One major issue is that pre-trained language models need more specific mathematical training, which can hinder their ability to perform math-related tasks as proficiently as natural language tasks. Large pre-trained models can be costly to train from scratch for specific tasks. Additionally, LLMs may need help with downstream tasks involving various input formats or modalities, such as structured tables [14].

Fortunately, researchers have developed a solution to these issues: The Chain-of-Thought (CoT)

Prompting method [7]. CoT uses intermediate natural language rationales as prompts to enable LLMs to generate reasoning chains and predict answers for input questions. CoT prompts can be especially helpful in solving math word problems by encouraging a step-by-step thought process [15]. Recent research has focused on enhancing CoT reasoning under the few-shot setting by selecting better in-context examples and creating more effective reasoning steps [16].

2.2. Process-based approaches

Process-based approaches optimize CoT reasoning for complex tasks. The least-to-most prompting method, proposed by [13], breaks down problems into sub-problems to solve them one by one. Khot et al. [17] used different prompts and diverse decomposition structures. Chen et al. [10] and Gao et al. [18] proposed a program-of-thoughts method that uses large language models to express reasoning as a program. Lu et al. [19] integrated different tools in Chameleon to improve the abilities of LLMs for compositional reasons.

Above all, while LLMs may face certain challenges with mathematical reasoning, researchers have developed effective methods to overcome them. With the help of CoT prompts and other techniques, LLMs can perform math-related tasks as well as natural language tasks.

3. The proposed method

The framework comprises two modules collaborating seamlessly to produce prompts, which assists the model in effectively reasoning complex problems: The MTM model utilizes a matrix of nouns and an adjective modifiers library to create a graph *NDG* linking to a knowledge base to solve problems. The knowledge base utilizes a known quantity and filters out the knowledge. The Figure 1 process is repeated until the objective uses a known quantity as a node. Finally, the solution engine is utilized to filter out the needed knowledge.

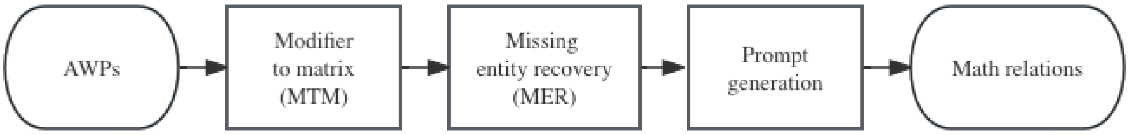
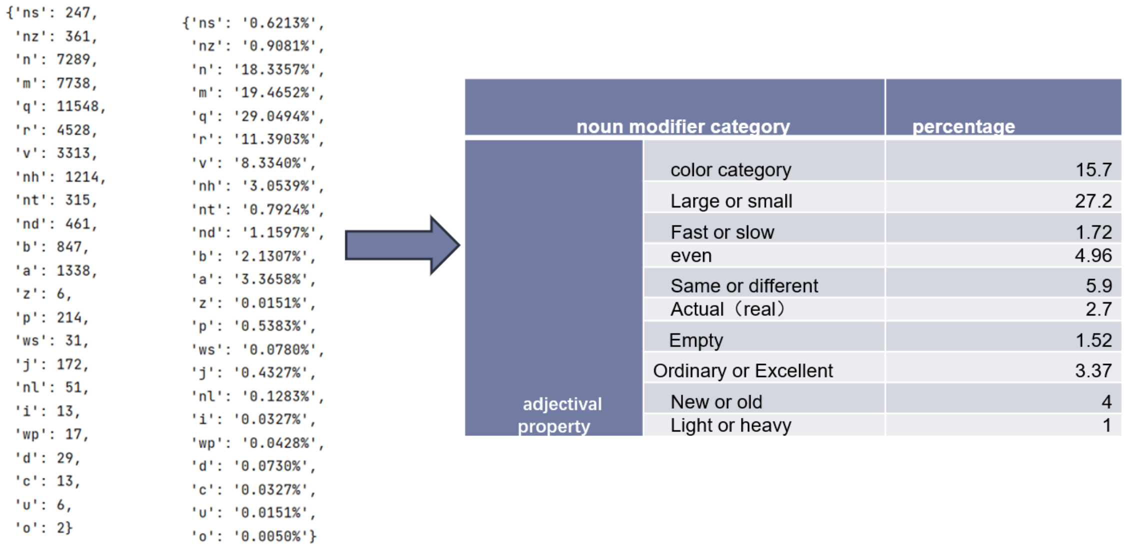


Figure 1. Flowchart of the method.

3.1. Modifier to matrix (MTM) model

The AWPs are annotated into a knowledge matrix with an explicit entity and a solution goal, and the method used to generate entity modifiers of AWPs has been effectively recategorized. The MTM extracts AWP implicit knowledge features and selects a commonsense and domain knowledge vocabulary from various learning resources. An MTM is centered information of an AWP syntax-semantic relation and connects the entities and attributes to the matrix. Through the graph matrix, we aggregate the contextual information of knowledge attribute words via Qualia role relation to form the knowledge representation of AWP. Finally, the AWP knowledge representation links implicit knowledge space using a learnable matrix.



```
{'ns': 247,
 'nz': 361,
 'n': 7289,
 'm': 7738,
 'q': 11548,
 'r': 4528,
 'v': 3313,
 'nh': 1214,
 'nt': 315,
 'nd': 461,
 'b': 847,
 'a': 1338,
 'z': 6,
 'p': 214,
 'ws': 31,
 'j': 172,
 'nl': 51,
 'i': 13,
 'wp': 17,
 'd': 29,
 'c': 13,
 'u': 6,
 'o': 2}
```

```
{'ns': '0.6213%',
 'nz': '0.9081%',
 'n': '18.3357%',
 'm': '19.4652%',
 'q': '29.0494%',
 'r': '11.3903%',
 'v': '8.3340%',
 'nh': '3.0539%',
 'nt': '0.7924%',
 'nd': '1.1597%',
 'b': '2.1307%',
 'a': '3.3658%',
 'z': '0.0151%',
 'p': '0.5383%',
 'ws': '0.0780%',
 'j': '0.4327%',
 'nl': '0.1283%',
 'i': '0.0327%',
 'wp': '0.0428%',
 'd': '0.0730%',
 'c': '0.0327%',
 'u': '0.0151%',
 'o': '0.0050%'}
```

noun modifier category		percentage
adjectival property	color category	15.7
	Large or small	27.2
	Fast or slow	1.72
	even	4.96
	Same or different	5.9
	Actual（real）	2.7
	Empty	1.52
	Ordinary or Excellent	3.37
	New or old	4
	Light or heavy	1

Figure 2. The proportion of each lexeme and the proportion of high-frequency adjectives.

```
Algorithm 1 Modifier word matrix construction algorithm
Require: The noun entity is noted as $x$, the noun in the $i$th position is noted as $x_i$, the jth Modifier in
    $x_i$ is noted as $y_{ij}$, the verb is noted as $v$, and the adjective is noted as $a$. $i, j = 1, 2, ..., m$.
Ensure: The set of elements of the word matrix, $M$
Set $R$ as empty. {A set of modifiers to be added.}
for each $x_i$ in $x$ && each $y_{ij}$ in $x_i$ do
    if $y_{ij}$ belongs to $v$ then
        if The agent is not in the agent list of $v$ then
            Add this agent to the agent list of $v$.
        end if
        if $y_{ij}$ belongs to a category of the set of adjectives then
            Add $y_{ij}$to this category list.
        end if
    end if
end for
for each element in all lists do
    if At least two elements in this list then
        Add this list to $R$
    end if
end for
Return $M$ =0
```

Figure 2 shows that the modifying component is adjectival. It accesses the adjective library to find adjectives and creates a collection of adjectival modifier classes for AWPs. The modifier has a distinguishing effect entity to understand the related knowledge attribute words and different scenarios. The knowledge attributes words in specific scenarios to others. The interrelation association weights, denoted as w_{ij}, calculate the connection between words a_i and scenarios s_j. These weights create a scene association matrix A, including the association weights between knowledge attribute words and scenarios. The number of knowledge scenarios k equals the number of knowledge points.

Algorithm 1 shows that the MTM model generated a word matrix, and each noun entity can also be generated with its corresponding word matrix representation according to the Qulia and syntax-semantic relation. The entities are consistent with previously generated noun entities. The modifications of these noun entities are redundant. Therefore, problem-solving could be modeled as path searching to connect the known nodes N to the unknown nodes in NDG until the goal g node is solved. Here, the goal node denotes the unknown variable asked to be solved in the question portion. The path searching determines nodes that could be selected to calculate an unknown node, especially in the case of implicit nodes in the graph. In the field of AWP solving, it can build a static NDG that contains domain knowledge and commonsense knowledge to store and represent the attribute relations of entities.

Besides, NDG can also represent the dynamic relations of entities given by the problem text. Ultimately, all the static and dynamic relations are integrated into a single NDG represented and stored the quantity relations.

For example: "There are three yellow apple trees planted by Xiao Ming, nine red apple trees planted by Xiao Ming and Xiao Li, eight apple trees planted by Xiao Ming, and the number of red apple trees planted by Xiao Li." The matrix of the "apple tree" is shown in the following table:

Table 1. Embedding matrix example.

apple trees	**Xiao Ming**	**Xiao Li**
yellow	X11	X12
red	X21	X22

The answers to the questions can be obtained in the order of reasoning:

Red apple tree planted by Xiao Ming = apple tree planted by Xiao Ming - yellow apple tree planted by Xiao Ming.

Red apple tree planted by Xiao Li = red apple tree planted by Xiao Ming and Xiao Li - red apple tree planted by Xiao Ming.

$$\begin{cases} a_2 = 9 \\ b_1 = 8 \\ x_{21} = 5 \\ x_{22} = 4 \end{cases} \tag{1}$$

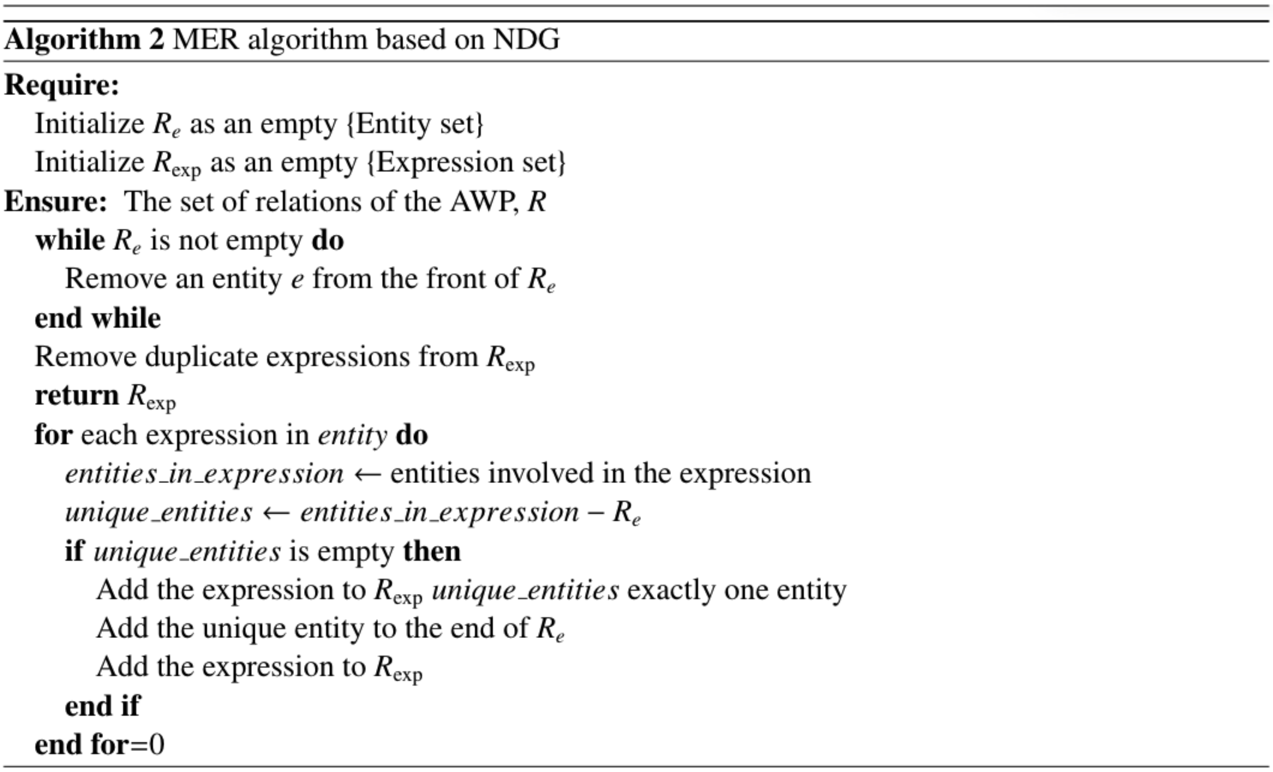
Extending to the general case, the data is handled in the following manner: The two modified components are denoted as M_{1n} and M_{2m}. The modified entities are then transformed into matrix form, which is illustrated in the table below. Here, *n* represents the row vector and *m* represents the column vector.

$$E_{m\times n} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} \tag{2}$$

The iterative equation system is comprised of three parts: Row sum, column sum, and total sum. It assigns the variables a to the row sum, b to the column sum, and c to the total sum. The equation can then be constructed using these variables as follows:

$$\sum_{j=1}^{n} x_{ij} = a_i \tag{3}$$

$$\sum_{i=1}^{m} x_{ij} = b_j \tag{4}$$



Algorithm 2 MER algorithm based on NDG

```
Require:
  Initialize $R_e$ as an empty {Entity set}
  Initialize $R_{\exp}$ as an empty {Expression set}
Ensure: The set of relations of the AWP, $R$
  while $R_e$ is not empty do
    Remove an entity $e$ from the front of $R_e$
  end while
  Remove duplicate expressions from $R_{\exp}$
  return $R_{\exp}$
  for each expression in $entity$ do
    $entities\_in\_expression \leftarrow$ entities involved in the expression
    $unique\_entities \leftarrow entities\_in\_expression - R_e$
    if $unique\_entities$ is empty then
      Add the expression to $R_{\exp}$ $unique\_entities$ exactly one entity
      Add the unique entity to the end of $R_e$
      Add the expression to $R_{\exp}$
    end if
  end for=0
```

3.2. Missing entity recovery (MER)

The MTM is created through statistical techniques to reclassify adjectives. The missing entity recovery pertains to the reasoning of attribute values that are necessary to comprehend the implicit knowledge within formulas and concepts of semantic features. The MER model categorizes relations into different types of entities and operations. By utilizing the *NDG* generation algorithm, mathematical entities can better infer implicit expressions and comprehend complex concepts and theories in mathematics more extensively.

The *NDG* represents entities and attributes values as nodes while organizing qualia relations as links in math word problem-solving. The *NDG* also contains domain knowledge and commonsense knowledge, which are essential in storing and representing attribute expressions of entities. As illustrated in Figure 3 and Figure 4, entity relationships can be classified into three fundamental types. This classification enables the organization of entity relationships into a network named *NDG*, which is composed of the three relationship networks mentioned above.

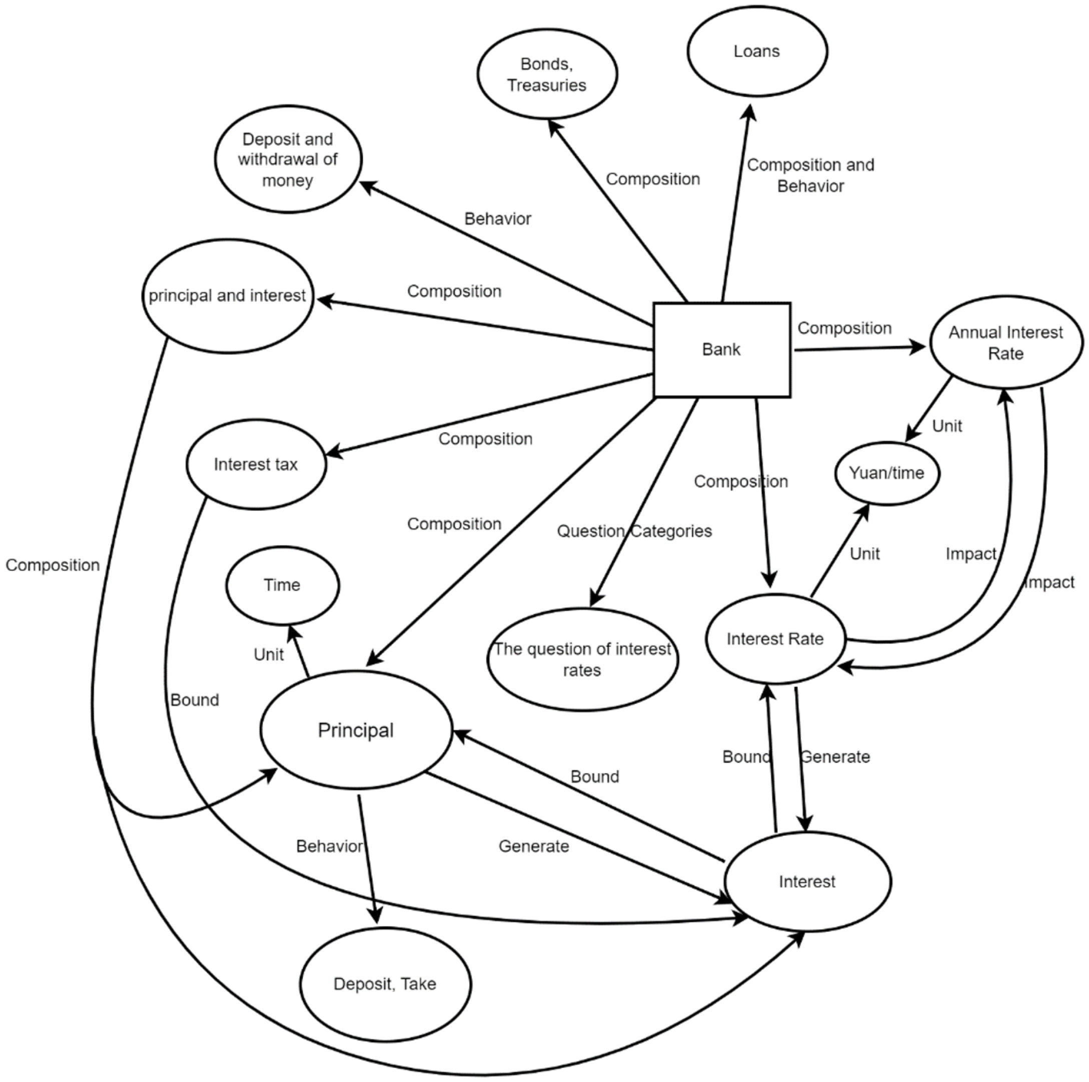


Figure 3. The NDG-back of interest problem.

In the implicit knowledge space, the feature acquisition process is strengthened by the vector knowledge attention mechanism represented by AWPs, which assigns adaptive weights by combining semantic and knowledge features. Our proposed MTM provides a robust solution for modeling from acquiring to generating auxiliary implicit relations and overcoming current drawbacks, such as the

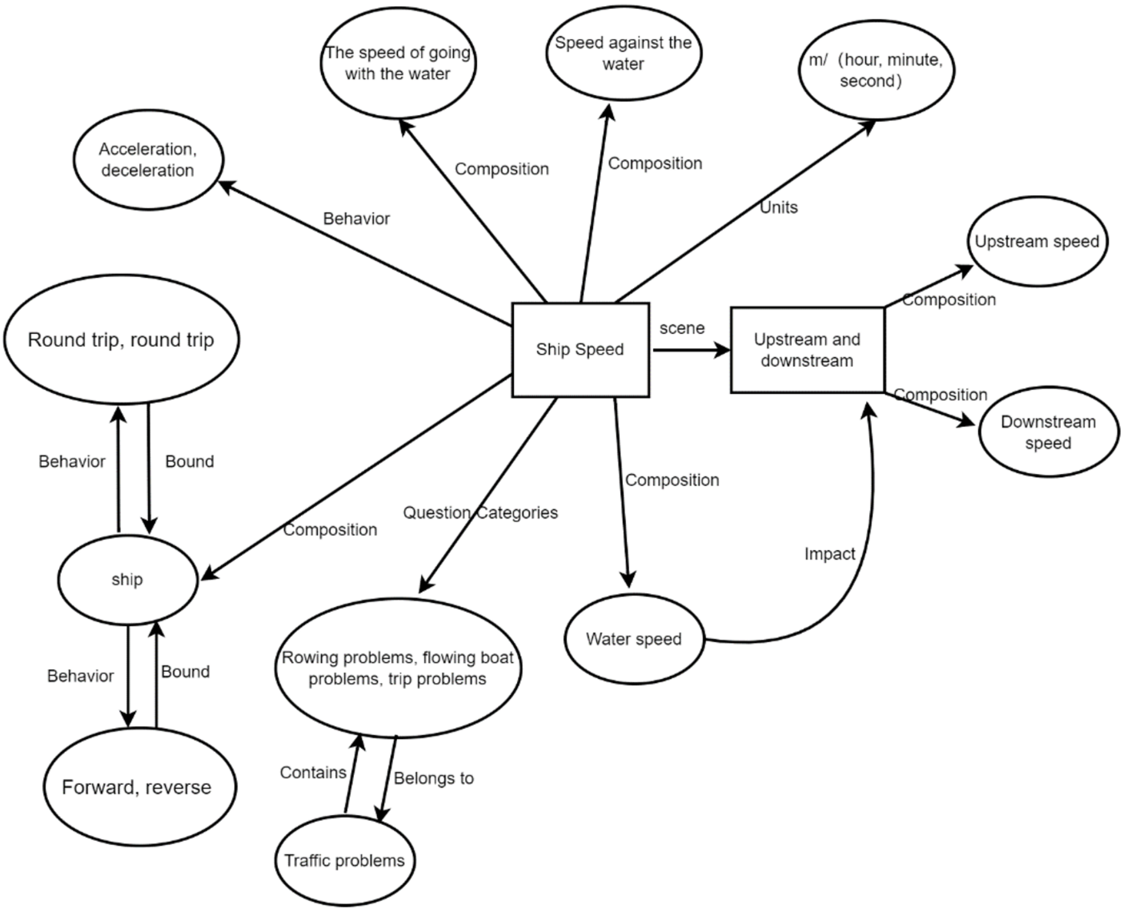


Figure 4. The NDG-back of traffic problem.

completeness of entity NDG relation generation and the assumed knowledge graph. This process is commonly referred to as an ontology-based auxiliary knowledge system during the code execution phase. Figures 3 and 4 provide compelling examples that demonstrate the effectiveness of the auxiliary generation of our model.

4. Experiment

This section presents the AWPs that correspond to the dataset type of high-profile AWPs. Following our proposed algorithmic model, these AWPs are submitted to the large language model to generate inferences for relations. The accuracy of the reasoning is then compared, considering whether it includes the logic generated by the recommendations from the chain of thought.

To evaluate the performance of our method, we collected a subset of AWPs from the most widely used and publicly available datasets in the field of machine-solving study. We then compared our results with other methods in the same field to demonstrate the effectiveness of our approach. Our evaluation was based on nine datasets, namely PEP, BNU, EEP, M23K, Ape5kT, MWP, USC, Dol2K, and Arith. Among these, three datasets - PEP, BNU, and USC - are derived from real textbooks. PEP contains AWPs from the 2018 edition of the textbook for elementary school students published by

Table 2. The distribution of dataset over two types of problems.

Datasets		**Explicit AWPs**	**Implicit AWPs**	**Problem collection source**
name	quantity	quantity	quantity	
PEP	504	226	278	Elementary Math Textbook
BNU	436	293	143	Elementary Math Textbook
EEP	2722	1641	1081	Primary Question Papers
M23K	20910	16568	4342	Data cleaned from math23k
Ape5KT	4570	3431	1139	Ape210k's test set
MWP	2373	2299	74	From MAWPS
USC	497	436	61	California Elementary Textbook
Dol2K	1878	1809	69	From Dolphin1878
Arith	1541	1053	488	From AllArith

Table 3. Large model inference comparison results.

	Reasoning accuracy		
Types of Large Models	**base**	Zero-shot-CoT	Our proposed prompt
ChatGLM-6b(int4)	3%	3.5%	4.5%
ChatGTP	63.5%	65%	68.5%

People's Education Press. BNU comprises arithmetic word problems from the 2018 edition of the textbook for elementary school students published by Beijing Normal University. The USC dataset includes arithmetic word problems from the California Elementary Mathematics Textbook. On the other hand, the EEP dataset comprises arithmetic word problems from the elementary school entrance examination papers of 34 provinces in China between 2010 and 2019. The remaining five datasets - M23K, Ape5kT, MWP, Dol2K, and Arith - are publicly available datasets in solution research.

The selection of a large model for our experiment has been limited, and we have chosen the ChatGLM-6B int4 version. This is an open-source and bilingual conversational language model based on the General Language Model (GLM) architecture with 6.2 billion parameters. With the help of model quantization techniques, it can be deployed locally on consumer-grade graphics cards. At the INT4 quantization level, it can be deployed on graphics cards with as low as 6GB of video memory. ChatGLM-6B uses a similar technology to ChatGPT, which is optimized for Chinese AWPs.

The inference ability of the large model can significantly vary based on various factors such as parameter magnitude and learning level. However, our proposed model has generated an effective prompt, which is evident from Table 2. We have found that including CoT in the large model for AWP problem reasoning can significantly enhance its reasoning ability. Furthermore, our improved methodology can further boost the reasoning ability of the humanoid. Based on the experimental results, we have found that the model's quality plays a crucial role in determining the reasoning ability of the large model. Our proposed method is a step forward in improving the reasoning ability of the model while keeping the model itself unchanged.

5. Conclusions

This paper proposed a generative model for explaining adjectival entity modifications and missing entity complementation to the large model prompt. The experiments conducted in this study show that the added prompt can effectively enhance the reasoning ability of the big model and help elementary and middle school students understand the questions and automatically generate a logical answer process.

While the model's implicit knowledge embedding has shown promise, there is still room for improvement to enhance its accuracy. Our plan is to continually expand the knowledge base and utilize vectorization techniques to improve word sense understanding. Currently, our model faces a specific error rate in selecting knowledge embeddings, which can be attributed to issues with retrieving. Nevertheless, with the development of a larger model and improvements to our existing one, we can significantly enhance the machine's reasoning expression ability. The latter will serve as the core technology for developing intelligent tutoring systems because it can potentially generate tutoring solutions.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments (All sources of funding of the study must be disclosed)

We would like to thank the constructive feedback provided by the reviewers. I gratefully acknowledge the financial support for my doctoral study provided by the China Scholarship Council (CSC). The Science and Technology Research Program of Chongqing supports this project. Municipal Education Commission (Grant No. KJZD-M202201801) and (Grant No. KJZD-K202001801).

Conflict of interest

The authors declare no conflicts of interest to report regarding the present study.

Ethics declaration

The author declared that the ethics committee approval was waived for the study.

References

1. Zhang, D., Wang, L., Zhang, L., Dai, B.T. and Shen, H.T., The gap of semantic parsing: A survey on automatic math word problem solvers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019, 42(9): 2287–2305. https://dx.doi.org/10.1109/TPAMI.2019.2914054
2. Sutskever, I., Vinyals, O. and Le, Q.V., Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 2014, 27. https://doi.org/10.48550/arXiv.1409.3215

3. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., et al., Attention is all you need. *Advances in neural information processing systems*, 2017, 30.

4. Ling, W., Yogatama, D., Dyer, C. and Blunsom, P., Program induction by rationale generation Learning to solve and explain algebraic word problems. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2017, 158–167. https://dx.doi.org/10.18653/v1/P17-1015

5. Yang, K. and Deng, J., Learning to prove theorems via interacting with proof assistants. *International Conference on Machine Learning (ICML)*, 2019, 97: 6984–6994.

6. Geva, M., Gupta, A. and Berant, J., Injecting numerical reasoning skills into language models. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020, 946–958.

7. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., et al., Chain of thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022, 35: 24824–24837.

8. Nakano, R., Hilton, J., Balaji, S., Wu, J., Ouyang, L., Kim, C., et al., Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

9. Zhou, D., Schärli, N., Hou, L., Wei, J., Scales, N., Wang, X., et al., Least-to-most prompting enables complex reasoning in large language models. *International Conference on Learning Representations (ICLR)*, 2023.

10. Chen, W., Ma, X., Wang, X. and Cohen, W.W., Program of thought prompting: Disentangling computation from reasoning for numerical reasoning tasks. *arXiv preprint arXiv:2211.12588*, 2022.

11. Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., Narang, S., et al., Self-consistency improves the chain of thought reasoning in language models. *International Conference on Learning Representations (ICLR)*, 2023.

12. Li, Y., Lin, Z., Zhang, S., Fu, Q., Chen, B., Lou, J.G., et al., On the advance of making language models better reasoners. *arXiv preprint arXiv:2206.02336*, 2022.

13. He, B., Meng, H., Zhang, Z., Liu, R., Zhang, T., Qualia Role-Based Quantity Relation Extraction for Solving Algebra Story Problems. *CMES-Computer Modeling in Engineering & Sciences*, 2023, 136(1): 403–419. https://doi.org/10.32604/cmes.2023.023242

14. Meng, H., Wu, H. and Yu, X., The Context-Oriented System Based on ELECTRA for Solving Math Word Problem. *2021 IEEE International Conference on Engineering, Technology & Education (TALE)*, 2021, 976–981. https://dx.doi.org/10.1109/TALE52509.2021.9678762

15. Zhao, Y., Li, Y., Li, C. and Zhang, R., Multihiertt: Numerical reasoning over multi-hierarchical tabular and textual data. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2022, 6588–6600. https://dx.doi.org/10.18653/v1/2022.acl-long.454

16. Kojima, T., Gu, S.S., Reid, M., Matsuo, Y. and Iwasawa, Y., Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 2022, 35: 22199–22213.

17. Khot, T., Trivedi, H., Finlayson, M., Fu, Y., Richardson, K., Clark, P., et al., Decomposed prompting: A modular approach for solving complex tasks. *arXiv preprint arXiv:2210.02406*, 2022.

18. Gao, L., Madaan, A., Zhou, S., Alon, U., Liu, P., Yang, Y., et al., Pal: Program-aided language models. *Proceedings of the 40th International Conference on Machine Learning*, 2023, 202: 10764–10799.

19. Lu, P., Peng, B., Cheng, H., Galley, M., Chang, K.W., Wu, Y.N., et al., Chameleon: Plug-and-play compositional reasoning with large language models. *arXiv preprint arXiv:2304.09842*, 2023.

Author’s biography

Mr. Hao Meng is a joint Ph.D. candidate in Education Technology at the Central China Normal University (CCNU) and the University of Wollongong (UOW). His research interests include Intelligent Tutoring Systems, Technology Enhanced Learning, and Automated Problem Solver.

Dr. Lin Yue is the Course Director of the Master of Business Analytics and Lecturer in the Department of Actuarial Studies and Business Analytics at Macquarie Business School. Her research is focused on intelligent networks, digital enablement, business / data analytics, strategic information systems and people analytics and modeling.

Dr. Geng Sun is an adjunct professor at the School of Engineering, Chongqing College of Humanities, Science and Technology, China and the director of Vermilion Cloud, Australia. His current research interests and focus are education in the Web3 era, AI in education, intelligent tutoring systems, and computational intelligence in adaptive learning.

Prof. Jun Shen is a Professor at the School of Computing and Information Technology, University of Wollongong, Wollongong, NSW, Australia. His expertise includes computational intelligence, bioinformatics, cloud computing, and learning technologies, including MOOC. He has published over 320 papers in journals and conferences in CS/IT areas.